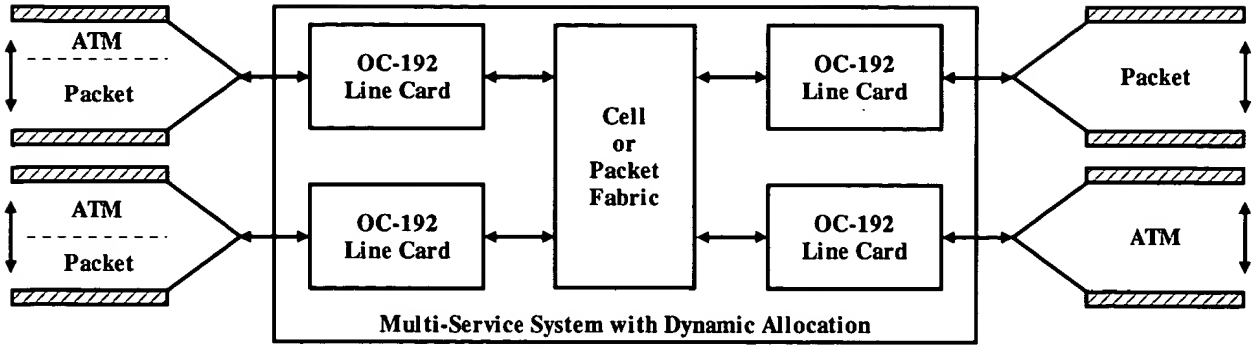


Figure 1

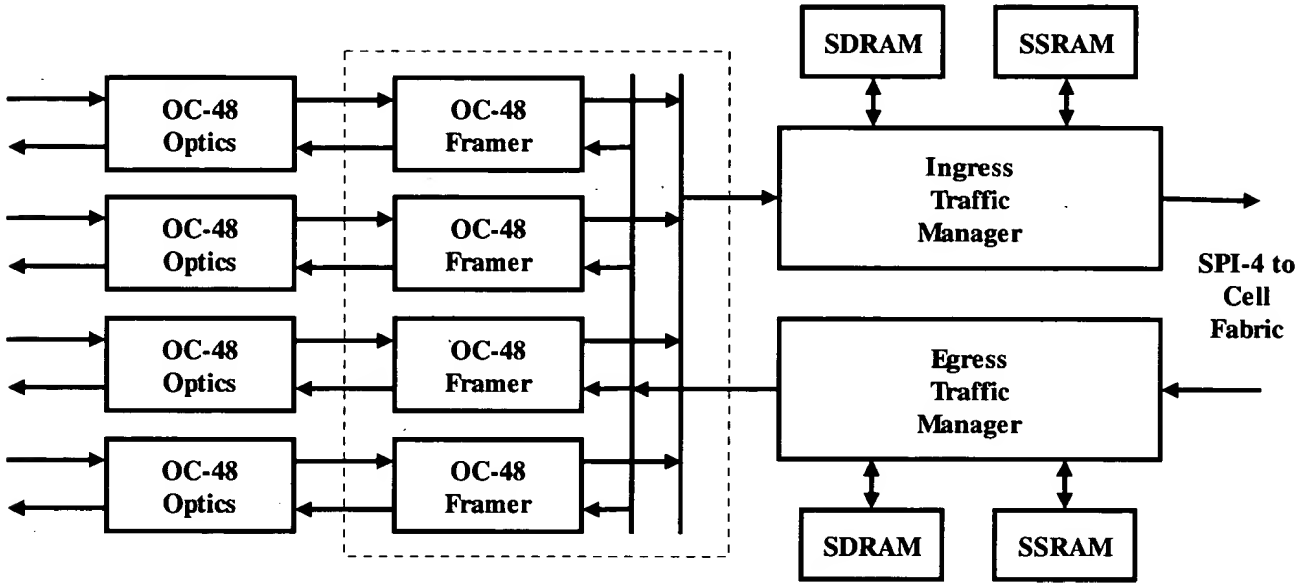


Figure 2

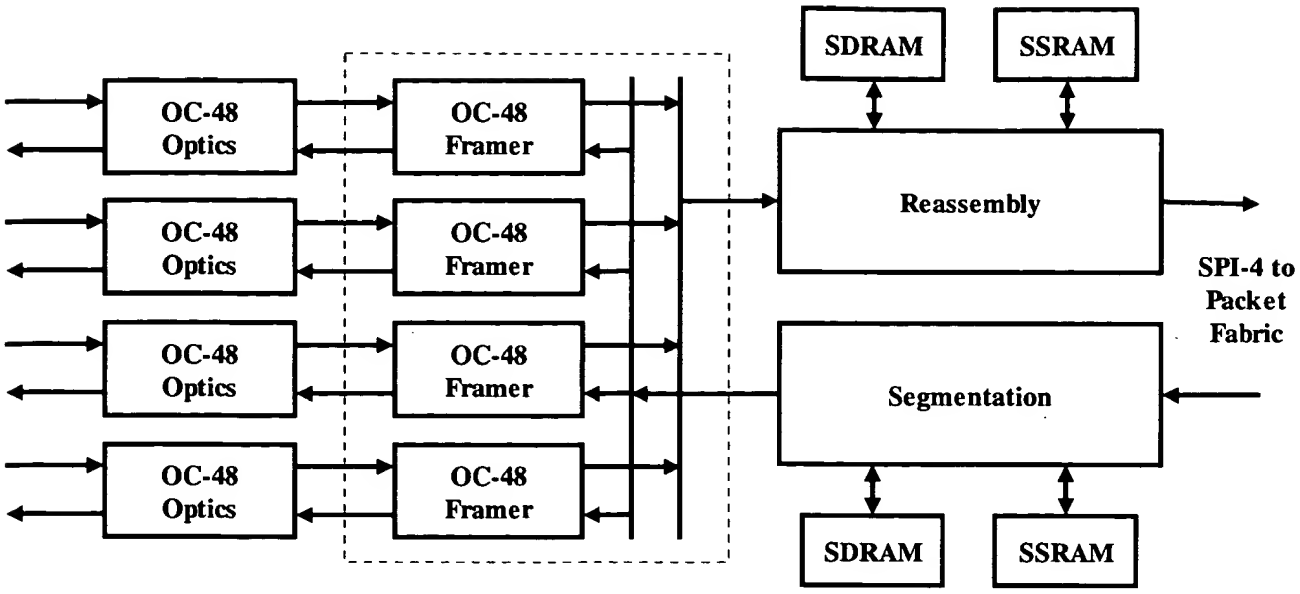


Figure 3

Type	Application	Incoming	Memory (64B Cell)	Outgoing
0	Ingress TM (ATM=>ATM)	ATM Cells	52B + 12B Pad	Switch Cell 8/16BHdr + 64B Cell
1	IngressTM (ATM⇒ MPLS Packet)	AAL5 Cells	48B + 16B Pad	
2	IngressTM (MPLS Packet⇒ATM)	Packet Bursts	48B + 16B Pad	
3	Ingress TM (Packet=>Packet)	Packet Bursts	64B Cells	
4	ATM Encapsulation	ATM Cells	52B + 12B Pad	0/8/16B Hdr + Full Packet
5	Reassembly	AAL5 Cells	48B + 16B Pad	
6	Ingress Packet Bypass	Packet Bursts	64B Cells	
7	Unused			
8	Egress TM (ATM=>ATM)	Switch Cell 8/16BHdr + 64B Cell	52B + 12B Pad	ATM Cells
9	EgressTM (ATM⇒ MPLS Packet)		48B + 16B Pad	Packet Bursts
10	EgressTM (MPLS Packet ⇒ATM)		48B + 16B Pad	AAL5 Cells
11	Egress TM (Packet=>Packet)		64B Cells	Packet Bursts
12	ATM De-Encapsulation	8/16B Hdr + Full Packet	52B + 12B Pad	ATM Cells
13	Segmentation		48B + 16B Pad	AAL5 Cells
14	Egress Packet Bypass		64B Cells	Packet Bursts
15	Unused			

Note: Incoming and Outgoing ATM Cells are non-AAL5 ATM cells for the purpose of Maximus chipset.

Figure 4

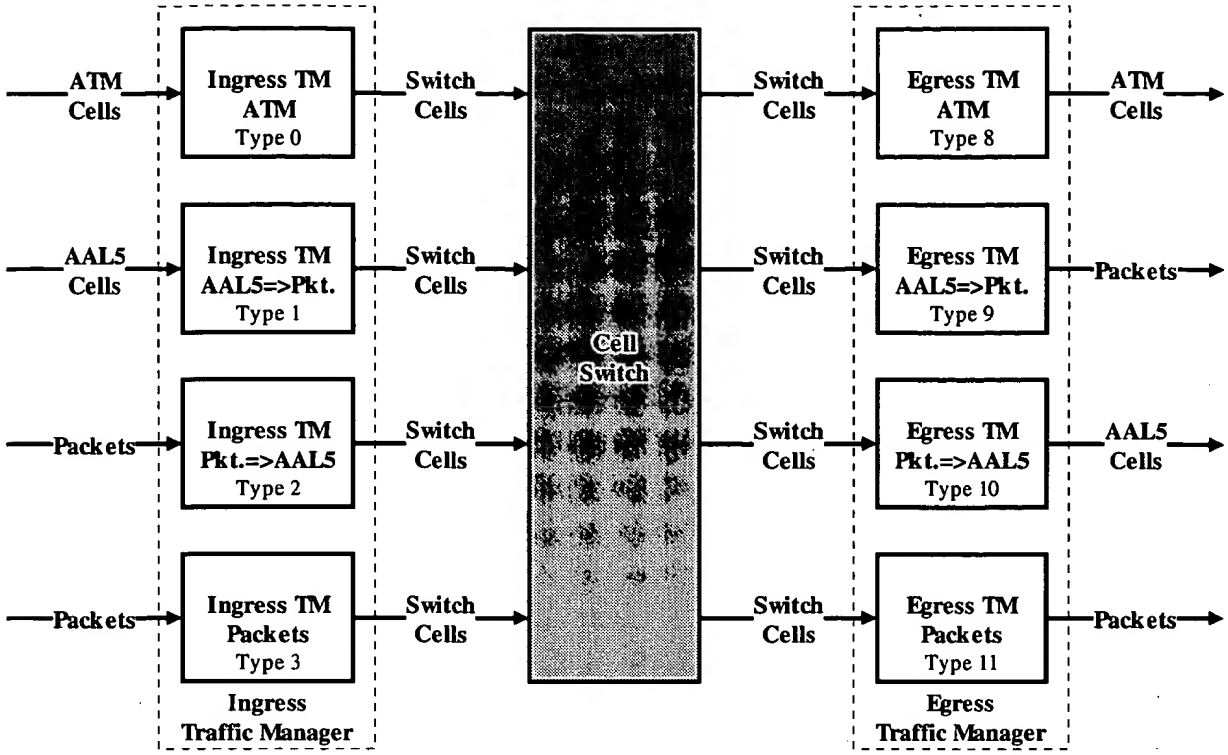


Figure 5

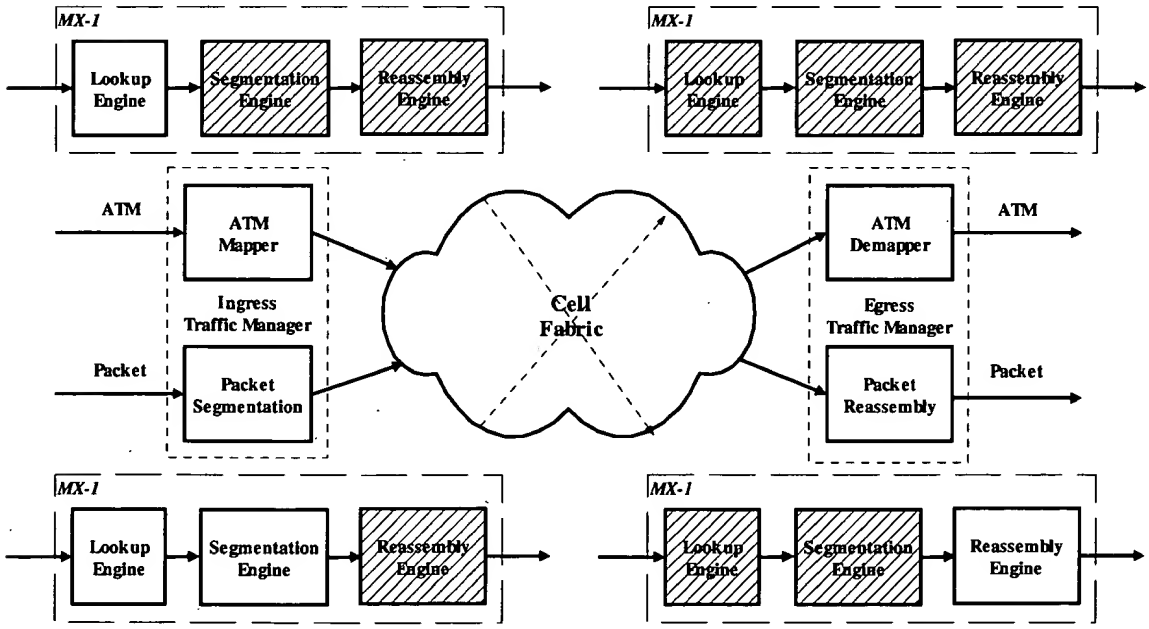


Figure 5A

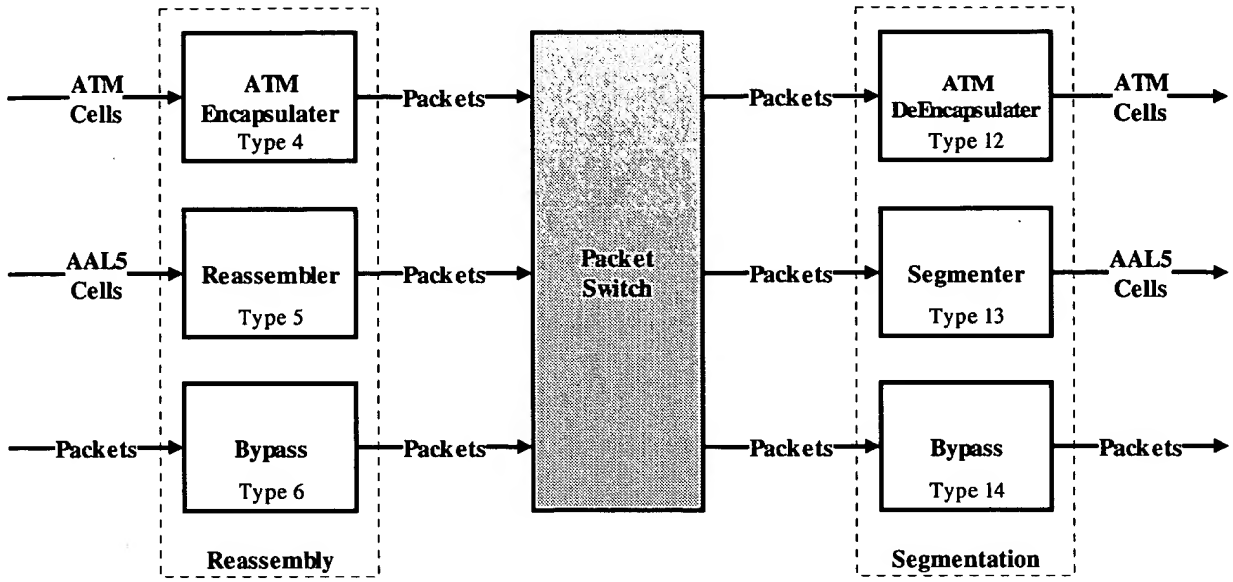


Figure 6

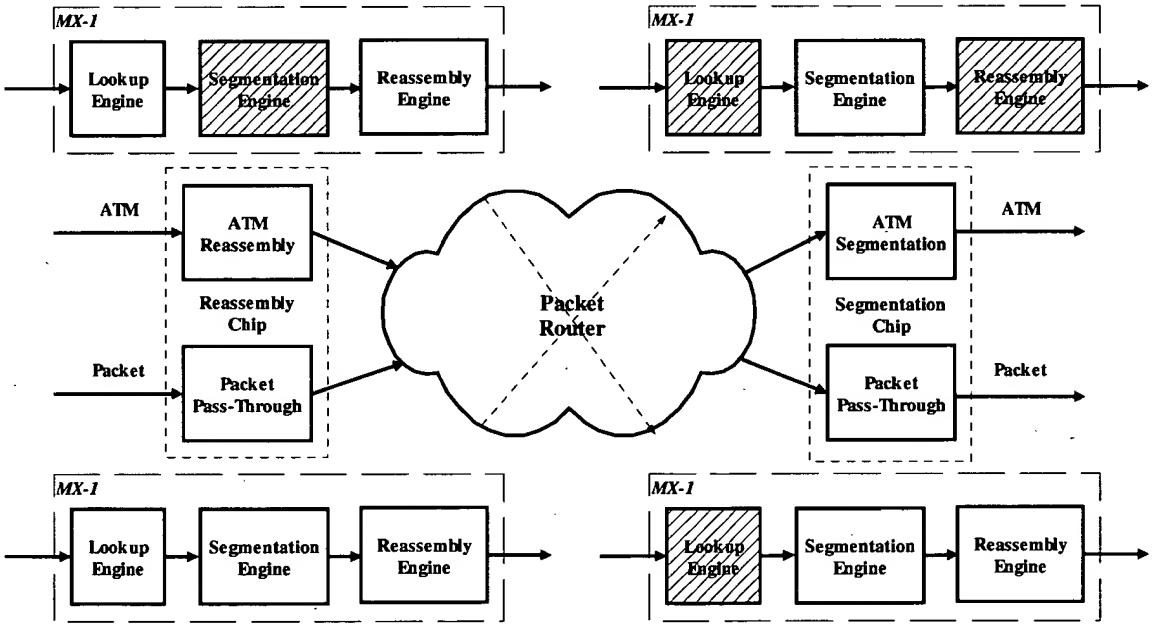


Figure 6A

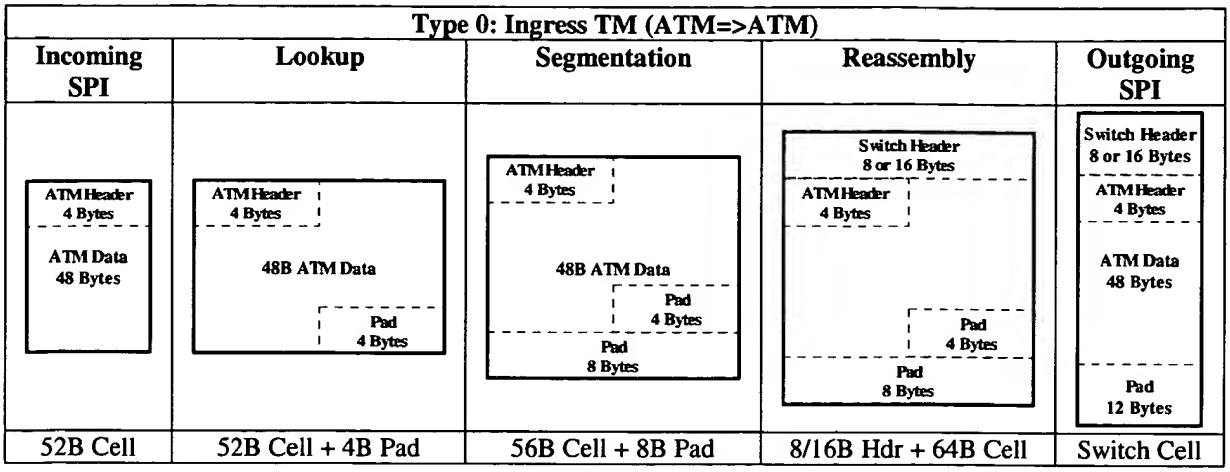


Figure 7

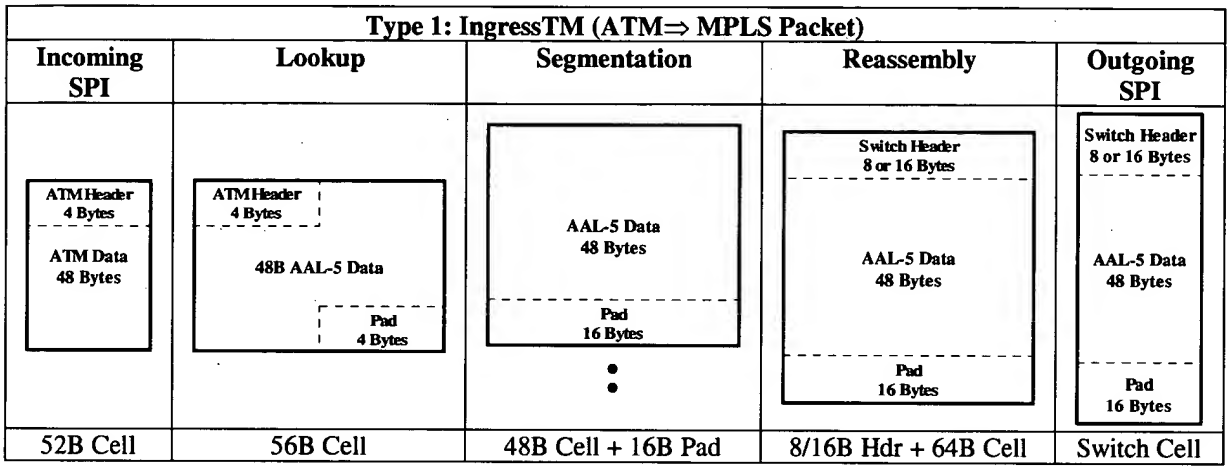


Figure 8

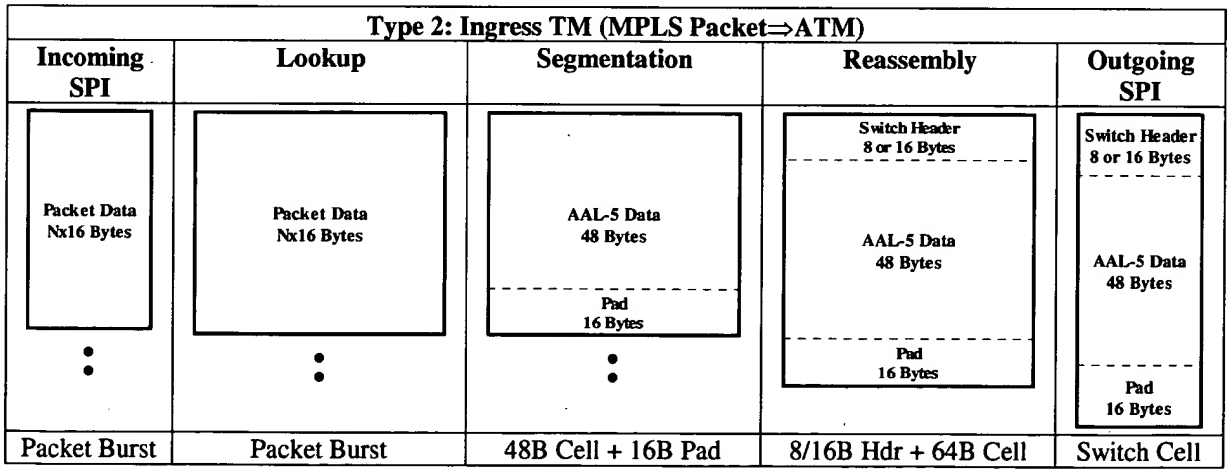


Figure 9

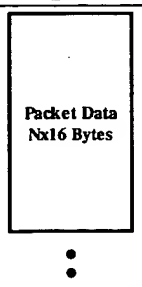
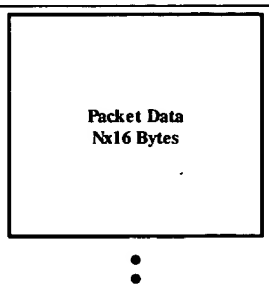
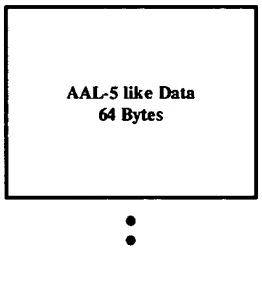
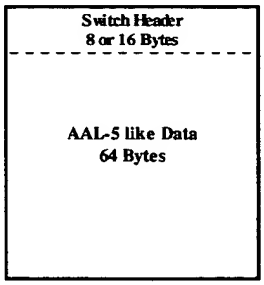
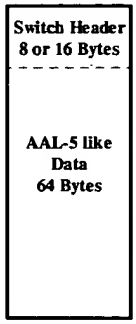
Type 3: Ingress TM (Packet=>Packet)				
Incoming SPI	**Lookup**	**Segmentation**	**Reassembly**	**Outgoing SPI**
				
Packet Burst	Packet Burst	AAL5 like 64B cell	8/16B Hdr + 64B Cell	Switch Cell

Figure 10

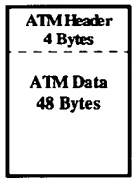
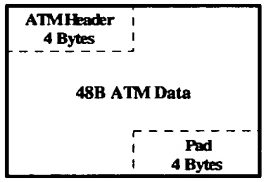
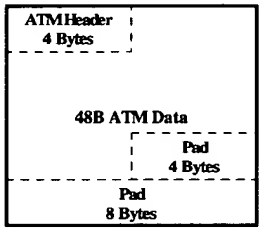
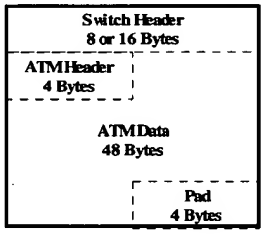
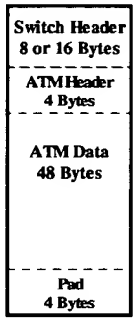
Type 4: ATM Encapsulation				
Incoming SPI	**Lookup**	**Segmentation**	**Reassembly**	**Outgoing SPI**
				
52B Cell + 4BPad	56B Cell	56B Cell + 8B Pad	8B Hdr + 52B Cell + 4B Pad	64B Packet

Figure 11

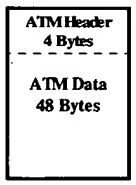
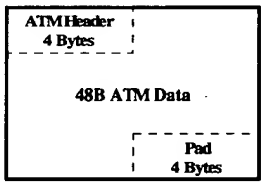
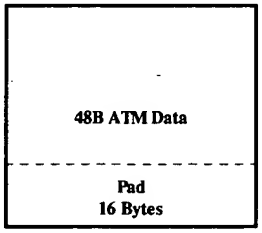
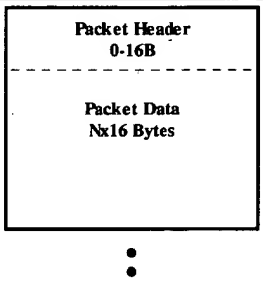
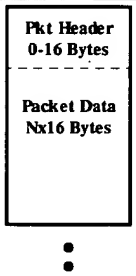
Type 5: Reassembly				
Incoming SPI	**Lookup**	**Segmentation**	**Reassembly**	**Outgoing SPI**
				
52B Cell + 4BPad	56B Cell	48B Cell + 16B Pad	0-16B Hdr + Packet Bursts	0-16B Hdr + Packet Bursts

Figure 12

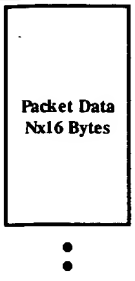
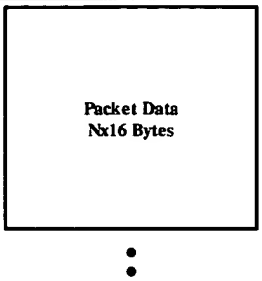
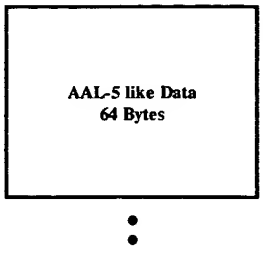
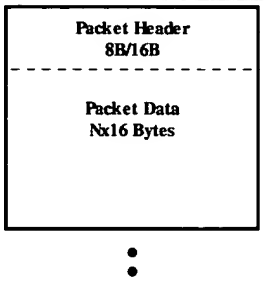
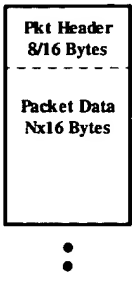
Type 6: Ingress Packet Bypass				
Incoming SPI	**Lookup**	**Segmentation**	**Reassembly**	**Outgoing SPI**
				
Packet Burst	Packet Burst	64B Cell (AAL5 like)	8B/16B Hdr + Packet Bursts	8/16 Hdr + Packet

Figure 13

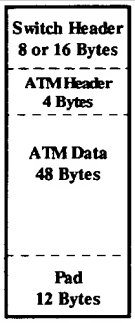
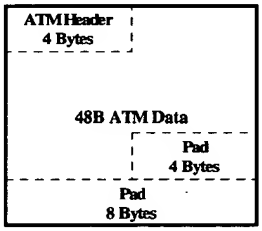
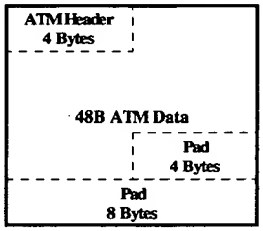
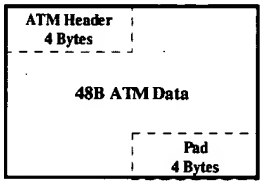
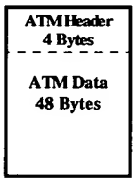
Type 8: Egress TM (ATM⇒ATM)				
Incoming SPI	**Lookup**	**Segmentation**	**Reassembly**	**Outgoing SPI**
				
Switch Cell	Strip switch header	64B Cell	Translate ATM Header	52B ATM Cell

Figure 14

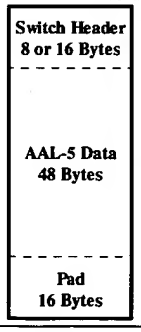
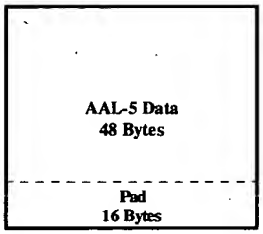
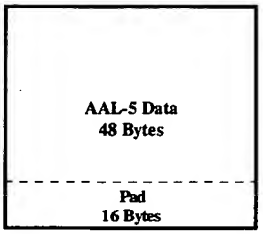
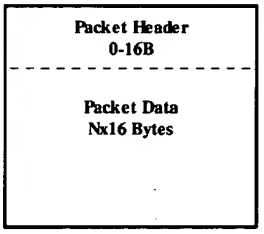
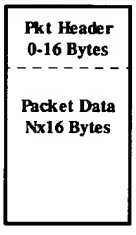
Type 9: EgressTM (ATM⇒MPLS Packet)				
Incoming SPI	**Lookup**	**Segmentation**	**Reassembly**	**Outgoing SPI**
				
52B Cell	56B Cell	48B Cell + 16B Pad	0-16B Hdr + Packet Bursts	0-16B Hdr + Packet

Figure 15

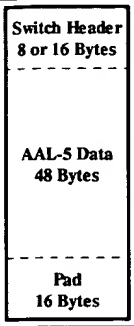
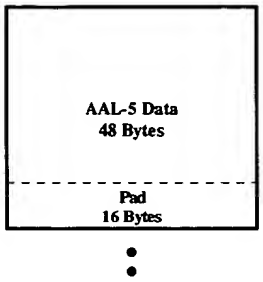
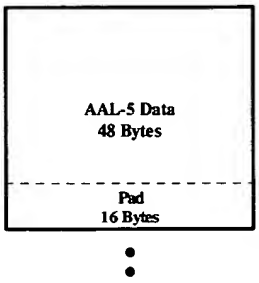
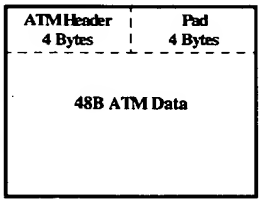
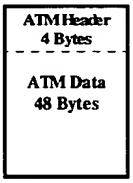
Type 10: EgressTM (MPLS Packet ⇒ATM)				
Incoming SPI	**Lookup**	**Segmentation**	**Reassembly**	**Outgoing SPI**
				
Packet Burst	Packet Burst	48B Cell + 16B Pad	Append ATM and 4B Pad, Strip 16B Pad	ATM Cell

Figure 16

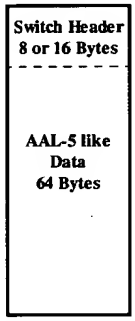
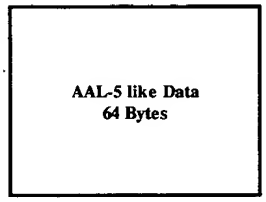
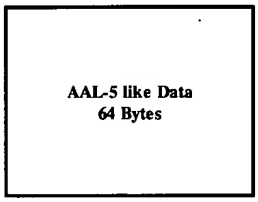
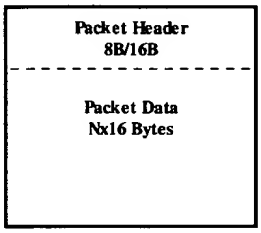
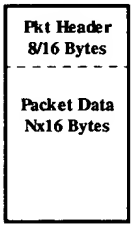
Type 11: Egress TM (Packet=>Packet)				
Incoming SPI	**Lookup**	**Segmentation**	**Reassembly**	**Outgoing SPI**
				
Packet Burst	Packet Burst	64B Cell (AAL5 like)	8B/16B Hdr + Packet Bursts	8/16 Hdr + Packet

Figure 17

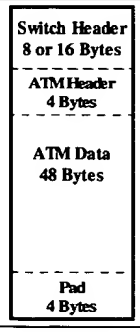
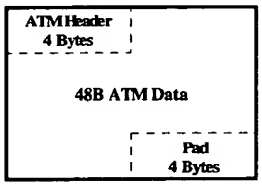
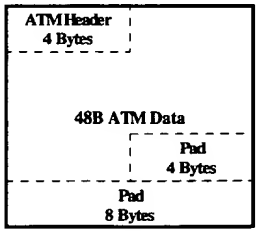
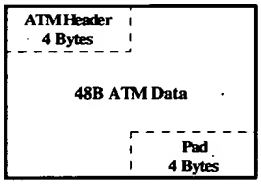
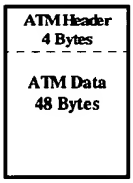
Type 12: ATM De-Encapsulation				
Incoming SPI	**Lookup**	**Segmentation**	**Reassembly**	**Outgoing SPI**
				
52B Cell + 4Bpad	56B Cell	56B Cell + 8B Pad	8B Hdr + 52B Cell + 4B Pad	ATM Cell

Figure 18

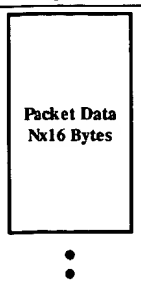
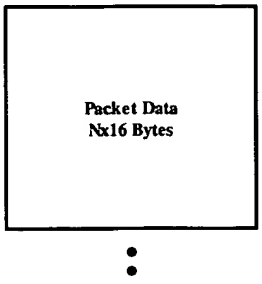
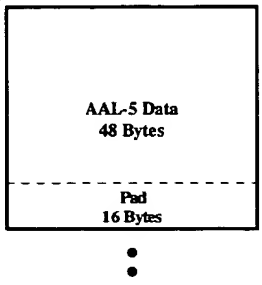
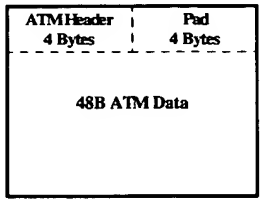
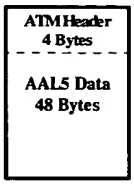
Type 13: Segmentation				
Incoming SPI	**Lookup**	**Segmentation**	**Reassembly**	**Outgoing SPI**
				
Packet Burst	Packet Burst	48B Cell + 16B Pad	Append ATM and 4B Pad, Strip 16B Pad	ATM Cell

Figure 19

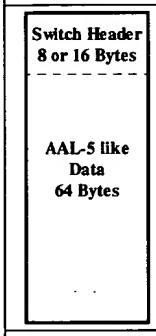
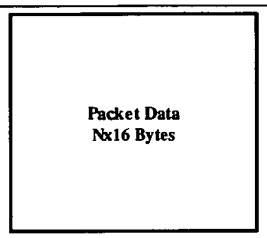
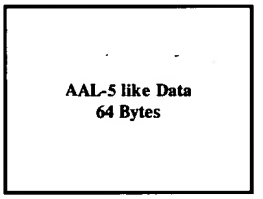
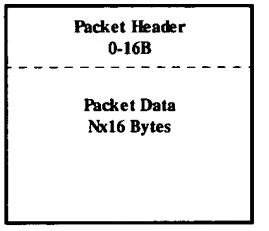
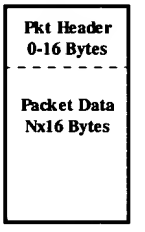
Type 14: Egress Packet Bypass				
Incoming SPI	**Lookup**	**Segmentation**	**Reassembly**	**Outgoing SPI**
				
Packet Burst	Packet Burst	64B Cell (AAL5 like)	0-16B Hdr + Packet Bursts	0-16B Hdr + Packet Bursts

Figure 20

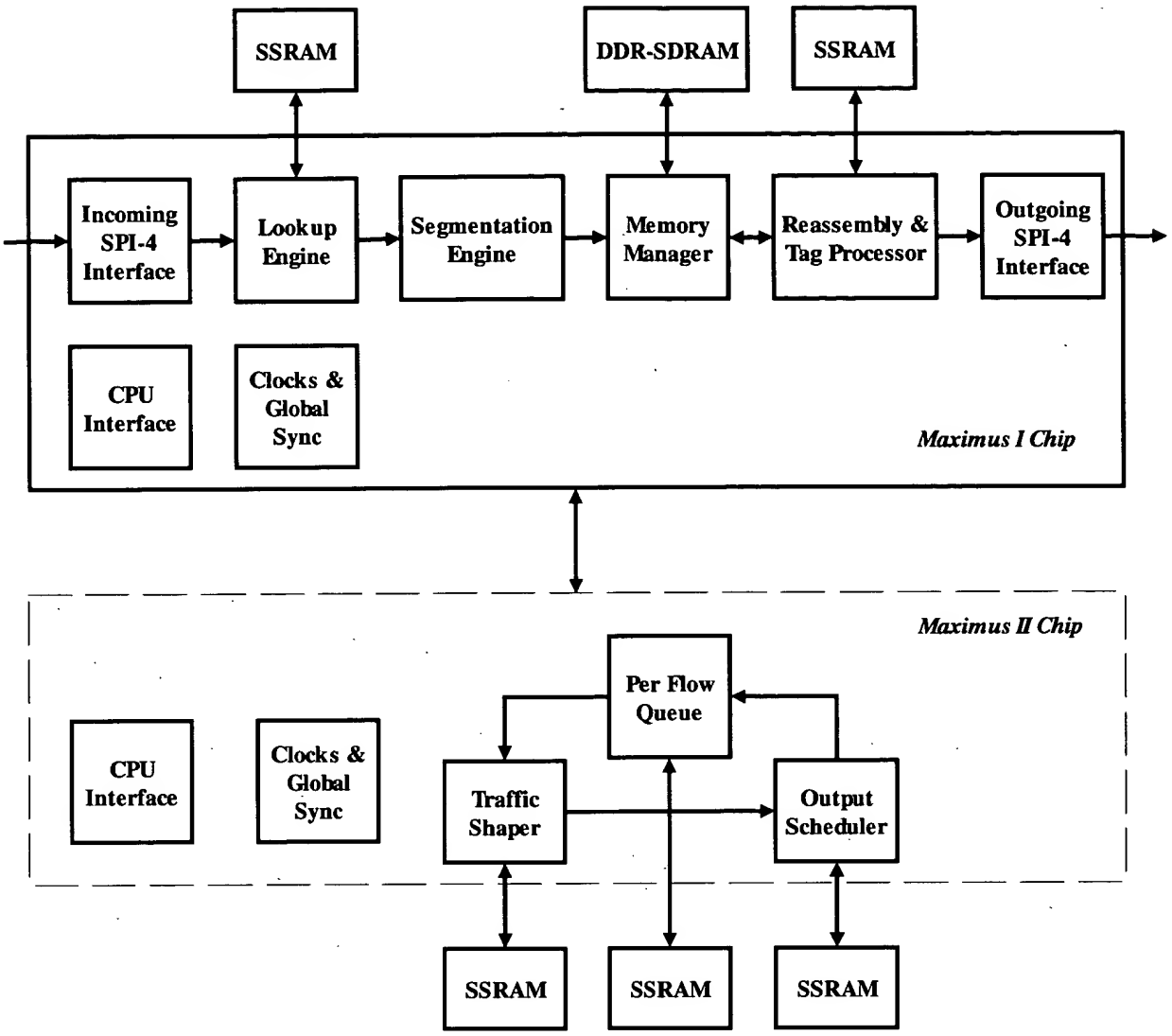

SSRAM
DDR-SDRAM
SSRAM
Incoming SPI-4 Interface
Lookup Engine
Segmentation Engine
Memory Manager
Reassembly & Tag Processor
Outgoing SPI-4 Interface
CPU Interface
Clocks & Global Sync
Maximus I Chip
Maximus II Chip
Per Flow Queue
CPU Interface
Clocks & Global Sync
Traffic Shaper
Output Scheduler
SSRAM
SSRAM
SSRAM

Figure 21

System						
Pin Name	**Pad#**	**Pin#**	**Dir**	**Input Buffer**	**Output Buffer**	**Description**
rst_l			IN			System reset
xtal_in_clk_200			IN			Crystal in or 200MHz input clock
xtal_out_clk			OUT			Crystal out
clk_400			IN			400MHz input clock
g_sync			IN			Global sync input
sys_config_2			IN			System configuration pin bit-2
sys_config_1			IN			System configuration pin bit-1
sys_config_0			IN			System configuration pin bit-0 000: Normal 001: SCAN 010: PLL test 011: JTAG 100: Test mux 101: BIST 110: reserved 111: reserved
Total pins = 8						
Incoming SPI_4						
Pin Name	**Pad#**	**Pin#**	**Dir**	**Input Buffer**	**Output Buffer**	**Description**
rdclk			IN			400MHz SPI receive clock
rdat_15			IN			SPI receive data bit-15
rdat_14			IN			SPI receive data bit-14
rdat_13			IN			SPI receive data bit-13
rdat_12			IN			SPI receive data bit-12
rdat_11			IN			SPI receive data bit-11
rdat_10			IN			SPI receive data bit-10
rdat_9			IN			SPI receive data bit-9
rdat_8			IN			SPI receive data bit-8
rdat_7			IN			SPI receive data bit-7
rdat_6			IN			SPI receive data bit-6
rdat_5			IN			SPI receive data bit-5
rdat_4			IN			SPI receive data bit-4
rdat_3			IN			SPI receive data bit-3
rdat_2			IN			SPI receive data bit-2
rdat_1			IN			SPI receive data bit-1
rdat_0			IN			SPI receive data bit-0

rdclk_l			IN			400MHz SPI receive clock
rdat_l_15			IN			SPI receive data bit-15
rdat_l_14			IN			SPI receive data bit-14
rdat_l_13			IN			SPI receive data bit-13
rdat_l_12			IN			SPI receive data bit-12
rdat_l_11			IN			SPI receive data bit-11
rdat_l_10			IN			SPI receive data bit-10
rdat_l_9			IN			SPI receive data bit-9
rdat_l_8			IN			SPI receive data bit-8
rdat_l_7			IN			SPI receive data bit-7
rdat_l_6			IN			SPI receive data bit-6
rdat_l_5			IN			SPI receive data bit-5
rdat_l_4			IN			SPI receive data bit-4
rdat_l_3			IN			SPI receive data bit-3
rdat_l_2			IN			SPI receive data bit-2
rdat_l_1			IN			SPI receive data bit-1
rdat_l_0			IN			SPI receive data bit-0
rctl			IN			SPI receive control input
rctl_l			IN			SPI receive control input
rsclk			IN			200MHz SPI receive status clock
rsclk_l			IN			200MHz SPI receive status clock
rstat_1			IN			SPI receive status bit-1
rstat_0			IN			SPI receive status bit-0
rstat_l_1			IN			SPI receive status bit-1
rstat_l_0			IN			SPI receive status bit-0
Total pins = 42						
Outgoing SPI_4						
Pin Name	**Pad#**	**Pin#**	**Dir**	**Input Buffer**	**Output Buffer**	**Description**
tdclk			OUT			400MHz SPI transmit clock
tdat_15			OUT			SPI transmit data bit-15
tdat_14			OUT			SPI transmit data bit-14
tdat_13			OUT			SPI transmit data bit-13
tdat_12			OUT			SPI transmit data bit-12
tdat_11			OUT			SPI transmit data bit-11
tdat_10			OUT			SPI transmit data bit-10
tdat_9			OUT			SPI transmit data bit-9
tdat_8			OUT			SPI transmit data bit-8
tdat_7			OUT			SPI transmit data bit-7

tdat_6			OUT			SPI transmit data bit-6
tdat_5			OUT			SPI transmit data bit-5
tdat_4			OUT			SPI transmit data bit-4
tdat_3			OUT			SPI transmit data bit-3
tdat_2			OUT			SPI transmit data bit-2
tdat_1			OUT			SPI transmit data bit-1
tdat_0			OUT			SPI transmit data bit-0
tdclk_1			OUT			400MHz SPI transmit clock
tdat_1_15			OUT			SPI transmit data bit-15
tdat_1_14			OUT			SPI transmit data bit-14
tdat_1_13			OUT			SPI transmit data bit-13
tdat_1_12			OUT			SPI transmit data bit-12
tdat_1_11			OUT			SPI transmit data bit-11
tdat_1_10			OUT			SPI transmit data bit-10
tdat_1_9			OUT			SPI transmit data bit-9
tdat_1_8			OUT			SPI transmit data bit-8
tdat_1_7			OUT			SPI transmit data bit-7
tdat_1_6			OUT			SPI transmit data bit-6
tdat_1_5			OUT			SPI transmit data bit-5
tdat_1_4			OUT			SPI transmit data bit-4
tdat_1_3			OUT			SPI transmit data bit-3
tdat_1_2			OUT			SPI transmit data bit-2
tdat_1_1			OUT			SPI transmit data bit-1
tdat_1_0			OUT			SPI transmit data bit-0
tctl			OUT			SPI transmit control input
tctl_1			OUT			SPI transmit control input
tsclk			OUT			200MHz SPI transmit status clock
tsclk_1			OUT			200MHz SPI transmit status clock
tstat_1			OUT			SPI transmit status bit-1
tstat_0			OUT			SPI transmit status bit-0
tstat_1_1			OUT			SPI transmit status bit-1
tstat_1_0			OUT			SPI transmit status bit-0
Output pins = 42						
spio_sch_clk			OUT			SPI to scheduler clock
spio_sch_sync			OUT			SPI to scheduler sync signal

spio_sch_status_8			OUT			SPI to scheduler status_8
spio_sch_status_7			OUT			SPI to scheduler status_7
spio_sch_status_6			OUT			SPI to scheduler status_6
spio_sch_status_5			OUT			SPI to scheduler status_5
spio_sch_status_4			OUT			SPI to scheduler status_4
spio_sch_status_3			OUT			SPI to scheduler status_3
spio_sch_status_2			OUT			SPI to scheduler status_2
spio_sch_status_1			OUT			SPI to scheduler status_1
spio_sch_status_0			OUT			SPI to scheduler status_0
Scheduler = 11 pins						
Total pins = 53						
Look Up Engine						
Pin Name	**Pad#**	**Pin#**	**Dir**	**Input Buffer**	**Output Buffer**	**Description**
lut_mem_i_clk			IN			
lut_mem_o_clk			OUT			
lut_mem1_we_l			OUT			Memory#1 write enable, active low
lut_mem1_addr_22			OUT			Memory#1 address bit-22
lut_mem1_addr_21			OUT			Memory#1 address bit-21
lut_mem1_addr_20			OUT			Memory#1 address bit-20
lut_mem1_addr_19			OUT			Memory#1 address bit-19
lut_mem1_addr_18			OUT			Memory#1 address bit-18
lut_mem1_addr_17			OUT			Memory#1 address bit-17
lut_mem1_addr_16			OUT			Memory#1 address bit-16
lut_mem1_addr_15			OUT			Memory#1 address bit-15
lut_mem1_addr_14			OUT			Memory#1 address bit-14

lut_mem1_addr_13			OUT			Memory#1 address bit-13
lut_mem1_addr_12			OUT			Memory#1 address bit-12
lut_mem1_addr_11			OUT			Memory#1 address bit-11
lut_mem1_addr_10			OUT			Memory#1 address bit-10
lut_mem1_addr_9			OUT			Memory#1 address bit-9
lut_mem1_addr_8			OUT			Memory#1 address bit-8
lut_mem1_addr_7			OUT			Memory#1 address bit-7
lut_mem1_addr_6			OUT			Memory#1 address bit-6
lut_mem1_addr_5			OUT			Memory#1 address bit-5
lut_mem1_addr_4			OUT			Memory#1 address bit-4
lut_mem1_addr_3			OUT			Memory#1 address bit-3
lut_mem1_addr_2			OUT			Memory#1 address bit-2
lut_mem1_addr_1			OUT			Memory#1 address bit-1
lut_mem1_addr_0			OUT			Memory#1 address bit-0
lut_mem1_data_63			BI			Memory#1 data bit-63
lut_mem1_data_62			BI			Memory#1 data bit-62
lut_mem1_data_61			BI			Memory#1 data bit-61
lut_mem1_data_60			BI			Memory#1 data bit-60
lut_mem1_data_59			BI			Memory#1 data bit-59
lut_mem1_data_58			BI			Memory#1 data bit-58
lut_mem1_data_57			BI			Memory#1 data bit-57
lut_mem1_data_56			BI			Memory#1 data bit-56
lut_mem1_data_55			BI			Memory#1 data bit-55
lut_mem1_data_54			BI			Memory#1 data bit-54
lut_mem1_data_53			BI			Memory#1 data bit-53
lut_mem1_data_52			BI			Memory#1 data bit-52
lut_mem1_data_51			BI			Memory#1 data bit-51
lut_mem1_data_50			BI			Memory#1 data bit-50
lut_mem1_data_49			BI			Memory#1 data bit-49
lut_mem1_data_48			BI			Memory#1 data bit-48
lut_mem1_data_47			BI			Memory#1 data bit-47
lut_mem1_data_46			BI			Memory#1 data bit-46

lut_mem1_data_45			BI			Memory#1 data bit-45
lut_mem1_data_44			BI			Memory#1 data bit-44
lut_mem1_data_43			BI			Memory#1 data bit-43
lut_mem1_data_42			BI			Memory#1 data bit-42
lut_mem1_data_41			BI			Memory#1 data bit-41
lut_mem1_data_40			BI			Memory#1 data bit-40
lut_mem1_data_39			BI			Memory#1 data bit-39
lut_mem1_data_38			BI			Memory#1 data bit-38
lut_mem1_data_37			BI			Memory#1 data bit-37
lut_mem1_data_36			BI			Memory#1 data bit-36
lut_mem1_data_35			BI			Memory#1 data bit-35
lut_mem1_data_34			BI			Memory#1 data bit-34
lut_mem1_data_33			BI			Memory#1 data bit-33
lut_mem1_data_32			BI			Memory#1 data bit-32
lut_mem1_data_31			BI			Memory#1 data bit-31
lut_mem1_data_30			BI			Memory#1 data bit-30
lut_mem1_data_29			BI			Memory#1 data bit-29
lut_mem1_data_28			BI			Memory#1 data bit-28
lut_mem1_data_27			BI			Memory#1 data bit-27
lut_mem1_data_26			BI			Memory#1 data bit-26
lut_mem1_data_25			BI			Memory#1 data bit-25
lut_mem1_data_24			BI			Memory#1 data bit-24
lut_mem1_data_23			BI			Memory#1 data bit-23
lut_mem1_data_22			BI			Memory#1 data bit-22
lut_mem1_data_21			BI			Memory#1 data bit-21
lut_mem1_data_20			BI			Memory#1 data bit-20
lut_mem1_data_19			BI			Memory#1 data bit-19
lut_mem1_data_18			BI			Memory#1 data bit-18
lut_mem1_data_17			BI			Memory#1 data bit-17
lut_mem1_data_16			BI			Memory#1 data bit-16
lut_mem1_data_15			BI			Memory#1 data bit-15
lut_mem1_data_14			BI			Memory#1 data bit-14
lut_mem1_data_13			BI			Memory#1 data bit-13
lut_mem1_data_12			BI			Memory#1 data bit-12
lut_mem1_data_11			BI			Memory#1 data bit-11
lut_mem1_data_10			BI			Memory#1 data bit-10
lut_mem1_data_9			BI			Memory#1 data bit-9
lut_mem1_data_8			BI			Memory#1 data bit-8
lut_mem1_data_7			BI			Memory#1 data bit-7
lut_mem1_data_6			BI			Memory#1 data bit-6
lut_mem1_data_5			BI			Memory#1 data bit-5
lut_mem1_data_4			BI			Memory#1 data bit-4
lut_mem1_data_3			BI			Memory#1 data bit-3
lut_mem1_data_2			BI			Memory#1 data bit-2
lut_mem1_data_1			BI			Memory#1 data bit-1

lut_mem1_data_0			BI			Memory#1 data bit-0
Memory#1 = 88 pins						
lut_mem2_we_l			OUT			Memory#2 write enable, active low
lut_mem2_addr_22			OUT			Memory#2 address bit-22
lut_mem2_addr_21			OUT			Memory#2 address bit-21
lut_mem2_addr_20			OUT			Memory#2 address bit-20
lut_mem2_addr_19			OUT			Memory#2 address bit-19
lut_mem2_addr_18			OUT			Memory#2 address bit-18
lut_mem2_addr_17			OUT			Memory#2 address bit-17
lut_mem2_addr_16			OUT			Memory#2 address bit-16
lut_mem2_addr_15			OUT			Memory#2 address bit-15
lut_mem2_addr_14			OUT			Memory#2 address bit-14
lut_mem2_addr_13			OUT			Memory#2 address bit-13
lut_mem2_addr_12			OUT			Memory#2 address bit-12
lut_mem2_addr_11			OUT			Memory#2 address bit-11
lut_mem2_addr_10			OUT			Memory#2 address bit-10
lut_mem2_addr_9			OUT			Memory#2 address bit-9
lut_mem2_addr_8			OUT			Memory#2 address bit-8
lut_mem2_addr_7			OUT			Memory#2 address bit-7
lut_mem2_addr_6			OUT			Memory#2 address bit-6
lut_mem2_addr_5			OUT			Memory#2 address bit-5
lut_mem2_addr_4			OUT			Memory#2 address bit-4
lut_mem2_addr_3			OUT			Memory#2 address bit-3
lut_mem2_addr_2			OUT			Memory#2 address bit-2

lut_mem2_addr_1			OUT			Memory#2 address bit-1
lut_mem2_addr_0			OUT			Memory#2 address bit-0
lut_mem2_data_63			BI			Memory#2 data bit-63
lut_mem2_data_62			BI			Memory#2 data bit-62
lut_mem2_data_61			BI			Memory#2 data bit-61
lut_mem2_data_60			BI			Memory#2 data bit-60
lut_mem2_data_59			BI			Memory#2 data bit-59
lut_mem2_data_58			BI			Memory#2 data bit-58
lut_mem2_data_57			BI			Memory#2 data bit-57
lut_mem2_data_56			BI			Memory#2 data bit-56
lut_mem2_data_55			BI			Memory#2 data bit-55
lut_mem2_data_54			BI			Memory#2 data bit-54
lut_mem2_data_53			BI			Memory#2 data bit-53
lut_mem2_data_52			BI			Memory#2 data bit-52
lut_mem2_data_51			BI			Memory#2 data bit-51
lut_mem2_data_50			BI			Memory#2 data bit-50
lut_mem2_data_49			BI			Memory#2 data bit-49
lut_mem2_data_48			BI			Memory#2 data bit-48
lut_mem2_data_47			BI			Memory#2 data bit-47
lut_mem2_data_46			BI			Memory#2 data bit-46
lut_mem2_data_45			BI			Memory#2 data bit-45
lut_mem2_data_44			BI			Memory#2 data bit-44
lut_mem2_data_43			BI			Memory#2 data bit-43
lut_mem2_data_42			BI			Memory#2 data bit-42
lut_mem2_data_41			BI			Memory#2 data bit-41
lut_mem2_data_40			BI			Memory#2 data bit-40
lut_mem2_data_39			BI			Memory#2 data bit-39
lut_mem2_data_38			BI			Memory#2 data bit-38
lut_mem2_data_37			BI			Memory#2 data bit-37
lut_mem2_data_36			BI			Memory#2 data bit-36
lut_mem2_data_35			BI			Memory#2 data bit-35
lut_mem2_data_34			BI			Memory#2 data bit-34
lut_mem2_data_33			BI			Memory#2 data bit-33
lut_mem2_data_32			BI			Memory#2 data bit-32
lut_mem2_data_31			BI			Memory#2 data bit-31
lut_mem2_data_30			BI			Memory#2 data bit-30
lut_mem2_data_29			BI			Memory#2 data bit-29
lut_mem2_data_28			BI			Memory#2 data bit-28
lut_mem2_data_27			BI			Memory#2 data bit-27
lut_mem2_data_26			BI			Memory#2 data bit-26
lut_mem2_data_25			BI			Memory#2 data bit-25
lut_mem2_data_24			BI			Memory#2 data bit-24
lut_mem2_data_23			BI			Memory#2 data bit-23

lut_mem2_data_22			BI			Memory#2 data bit-22
lut_mem2_data_21			BI			Memory#2 data bit-21
lut_mem2_data_20			BI			Memory#2 data bit-20
lut_mem2_data_19			BI			Memory#2 data bit-19
lut_mem2_data_18			BI			Memory#2 data bit-18
lut_mem2_data_17			BI			Memory#2 data bit-17
lut_mem2_data_16			BI			Memory#2 data bit-16
lut_mem2_data_15			BI			Memory#2 data bit-15
lut_mem2_data_14			BI			Memory#2 data bit-14
lut_mem2_data_13			BI			Memory#2 data bit-13
lut_mem2_data_12			BI			Memory#2 data bit-12
lut_mem2_data_11			BI			Memory#2 data bit-11
lut_mem2_data_10			BI			Memory#2 data bit-10
lut_mem2_data_9			BI			Memory#2 data bit-9
lut_mem2_data_8			BI			Memory#2 data bit-8
lut_mem2_data_7			BI			Memory#2 data bit-7
lut_mem2_data_6			BI			Memory#2 data bit-6
lut_mem2_data_5			BI			Memory#2 data bit-5
lut_mem2_data_4			BI			Memory#2 data bit-4
lut_mem2_data_3			BI			Memory#2 data bit-3
lut_mem2_data_2			BI			Memory#2 data bit-2
lut_mem2_data_1			BI			Memory#2 data bit-1
lut_mem2_data_0			BI			Memory#2 data bit-0
Memory#2 = 88 pins						
Total pins = 178						
Segmentation Engine						
Pin Name	**Pad#**	**Pin#**	**Dir**	**Input Buffer**	**Output Buffer**	**Description**
pfq_seg_clk			IN			PFQ=>SEG clock
pfq_seg_valid			IN			PFQ=>SEG valid signal
pfq_seg_data			IN			PFQ=>SEG bit data
seg_sch_clk			OUT			SEG=>SCH clock
seg_sch_valid			OUT			SEG=>SCH valid signal
seg_sch_data			OUT			SEG=>SCH bit data
Total pins = 6						
Memory Manager						
Pin Name	**Pad#**	**Pin#**	**Dir**	**Input Buffer**	**Output Buffer**	**Description**
mem_sdr_data_a_64			BI			Controller A data bus bit-64
mem_sdr_data_a_63			BI			Controller A data bus bit-63

mem_sdr_data_a_62			BI			Controller A data bus bit-62
mem_sdr_data_a_61			BI			Controller A data bus bit-61
mem_sdr_data_a_60			BI			Controller A data bus bit-60
mem_sdr_data_a_59			BI			Controller A data bus bit-59
mem_sdr_data_a_58			BI			Controller A data bus bit-58
mem_sdr_data_a_57			BI			Controller A data bus bit-57
mem_sdr_data_a_56			BI			Controller A data bus bit-56
mem_sdr_data_a_55			BI			Controller A data bus bit-55
mem_sdr_data_a_54			BI			Controller A data bus bit-54
mem_sdr_data_a_53			BI			Controller A data bus bit-53
mem_sdr_data_a_52			BI			Controller A data bus bit-52
mem_sdr_data_a_51			BI			Controller A data bus bit-51
mem_sdr_data_a_50			BI			Controller A data bus bit-50
mem_sdr_data_a_49			BI			Controller A data bus bit-49
mem_sdr_data_a_48			BI			Controller A data bus bit-48
mem_sdr_data_a_47			BI			Controller A data bus bit-47
mem_sdr_data_a_46			BI			Controller A data bus bit-46
mem_sdr_data_a_45			BI			Controller A data bus bit-45
mem_sdr_data_a_44			BI			Controller A data bus bit-44
mem_sdr_data_a_43			BI			Controller A data bus bit-43
mem_sdr_data_a_49			BI			Controller A data bus bit-42
mem_sdr_data_a_41			BI			Controller A data bus bit-41
mem_sdr_data_a_40			BI			Controller A data bus bit-40

mem_sdr_data_a_39			BI			Controller A data bus bit-39
mem_sdr_data_a_38			BI			Controller A data bus bit-38
mem_sdr_data_a_37			BI			Controller A data bus bit-37
mem_sdr_data_a_36			BI			Controller A data bus bit-36
mem_sdr_data_a_35			BI			Controller A data bus bit-35
mem_sdr_data_a_34			BI			Controller A data bus bit-34
mem_sdr_data_a_33			BI			Controller A data bus bit-33
mem_sdr_data_a_32			BI			Controller A data bus bit-32
mem_sdr_data_a_31			BI			Controller A data bus bit-31
mem_sdr_data_a_30			BI			Controller A data bus bit-30
mem_sdr_data_a_29			BI			Controller A data bus bit-29
mem_sdr_data_a_28			BI			Controller A data bus bit-28
mem_sdr_data_a_27			BI			Controller A data bus bit-27
mem_sdr_data_a_26			BI			Controller A data bus bit-26
mem_sdr_data_a_25			BI			Controller A data bus bit-25
mem_sdr_data_a_24			BI			Controller A data bus bit-24
mem_sdr_data_a_23			BI			Controller A data bus bit-23
mem_sdr_data_a_22			BI			Controller A data bus bit-22
mem_sdr_data_a_21			BI			Controller A data bus bit-22
mem_sdr_data_a_20			BI			Controller A data bus bit-20
mem_sdr_data_a_19			BI			Controller A data bus bit-19
mem_sdr_data_a_18			BI			Controller A data bus bit-18
mem_sdr_data_a_17			BI			Controller A data bus bit-17

mem_sdr_data_a_16			BI			Controller A data bus bit-16
mem_sdr_data_a_15			BI			Controller A data bus bit-15
mem_sdr_data_a_14			BI			Controller A data bus bit-14
mem_sdr_data_a_13			BI			Controller A data bus bit-13
mem_sdr_data_a_12			BI			Controller A data bus bit-12
mem_sdr_data_a_11			BI			Controller A data bus bit-11
mem_sdr_data_a_10			BI			Controller A data bus bit-10
mem_sdr_data_a_9			BI			Controller A data bus bit-9
mem_sdr_data_a_8			BI			Controller A data bus bit-8
mem_sdr_data_a_7			BI			Controller A data bus bit-7
mem_sdr_data_a_6			BI			Controller A data bus bit-6
mem_sdr_data_a_5			BI			Controller A data bus bit-5
mem_sdr_data_a_4			BI			Controller A data bus bit-4
mem_sdr_data_a_3			BI			Controller A data bus bit-3
mem_sdr_data_a_2			BI			Controller A data bus bit-2
mem_sdr_data_a_1			BI			Controller A data bus bit-1
mem_sdr_data_a_0			BI			Controller A data bus bit-0
mem_sdr_addr0_a_11			OUT			Controller A addr0 bit-11
mem_sdr_addr0_a_10			OUT			Controller A addr0 bit-10
mem_sdr_addr0_a_9			OUT			Controller A addr0 bit-9
mem_sdr_addr0_a_8			OUT			Controller A addr0 bit-8
mem_sdr_addr0_a_7			OUT			Controller A addr0 bit-7
mem_sdr_addr0_a_6			OUT			Controller A addr0 bit-6

mem_sdr_addr0_a_5			OUT			Controller A addr0 bit-5
mem_sdr_addr0_a_4			OUT			Controller A addr0 bit-4
mem_sdr_addr0_a_3			OUT			Controller A addr0 bit-3
mem_sdr_addr0_a_2			OUT			Controller A addr0 bit-2
mem_sdr_addr0_a_1			OUT			Controller A addr0 bit-1
mem_sdr_addr0_a_0			OUT			Controller A addr0 bit-0
mem_sdr_addr1_a_11			OUT			Controller A addr1 bit-11
mem_sdr_addr1_a_10			OUT			Controller A addr1 bit-10
mem_sdr_addr1_a_9			OUT			Controller A addr1 bit-9
mem_sdr_addr1_a_8			OUT			Controller A addr1 bit-8
mem_sdr_addr1_a_7			OUT			Controller A addr1 bit-7
mem_sdr_addr1_a_6			OUT			Controller A addr1 bit-6
mem_sdr_addr1_a_5			OUT			Controller A addr1 bit-5
mem_sdr_addr1_a_4			OUT			Controller A addr1 bit-4
mem_sdr_addr1_a_3			OUT			Controller A addr1 bit-3
mem_sdr_addr1_a_2			OUT			Controller A addr1 bit-2
mem_sdr_addr1_a_1			OUT			Controller A addr1 bit-1
mem_sdr_addr1_a_0			OUT			Controller A addr1 bit-0
mem_sdr_bank0_a_1			OUT			Controller A bank0 select bit-1
mem_sdr_bank0_a_0			OUT			Controller A bank0 select bit-0
mem_sdr_bank1_a_1			OUT			Controller A bank1 select bit-1
mem_sdr_bank1_a_0			OUT			Controller A bank1 select bit-0
mem_sdr_cs_a_7			OUT			Controller A chip select bit-7

mem_sdr_cs_a_6			OUT			Controller A chip select bit-6
mem_sdr_cs_a_5			OUT			Controller A chip select bit-5
mem_sdr_cs_a_4			OUT			Controller A chip select bit-4
mem_sdr_cs_a_3			OUT			Controller A chip select bit-3
mem_sdr_cs_a_2			OUT			Controller A chip select bit-2
mem_sdr_cs_a_1			OUT			Controller A chip select bit-1
mem_sdr_cs_a_0			OUT			Controller A chip select bit-0
mem_sdr_dqs0_a_1			BI			Controller A data strobe0 bit-1
mem_sdr_dqs0_a_0			BI			Controller A data strobe0 bit-0
mem_sdr_dqs1_a_1			BI			Controller A data strobe1 bit-1
mem_sdr_dqs1_a_0			BI			Controller A data strobe1 bit-0
mem_sdr_ras0_a			OUT			Controller A RAS0
mem_sdr_cas0_a			OUT			Controller A CAS0
mem_sdr_we0_a			OUT			Controller A write enable0
mem_sdr_clk0_a			OUT			Controller A differential clock output0
mem_sdr_clk0_a_l			OUT			Controller A differential clock output0, low active
mem_sdr_ras1_a			OUT			Controller A RAS1
mem_sdr_cas1_a			OUT			Controller A CAS1
mem_sdr_we1_a			OUT			Controller A write enable1
mem_sdr_clk1_a			OUT			Controller A differential clock output1
mem_sdr_clk1_a_l			OUT			Controller A differential clock output1, low active
mem_sdr_clke_a			OUT			Controller A clock enable
Controller A = 115 pins						

mem_sdr_data_b_64			BI			Controller B data bus bit-64
mem_sdr_data_b_63			BI			Controller B data bus bit-63
mem_sdr_data_b_62			BI			Controller B data bus bit-62
mem_sdr_data_b_61			BI			Controller B data bus bit-61
mem_sdr_data_b_60			BI			Controller B data bus bit-60
mem_sdr_data_b_59			BI			Controller B data bus bit-59
mem_sdr_data_b_58			BI			Controller B data bus bit-58
mem_sdr_data_b_57			BI			Controller B data bus bit-57
mem_sdr_data_b_56			BI			Controller B data bus bit-56
mem_sdr_data_b_55			BI			Controller B data bus bit-55
mem_sdr_data_b_54			BI			Controller B data bus bit-54
mem_sdr_data_b_53			BI			Controller B data bus bit-53
mem_sdr_data_b_52			BI			Controller B data bus bit-52
mem_sdr_data_b_51			BI			Controller B data bus bit-51
mem_sdr_data_b_50			BI			Controller B data bus bit-50
mem_sdr_data_b_49			BI			Controller B data bus bit-49
mem_sdr_data_b_48			BI			Controller B data bus bit-48
mem_sdr_data_b_47			BI			Controller B data bus bit-47
mem_sdr_data_b_46			BI			Controller B data bus bit-46
mem_sdr_data_b_45			BI			Controller B data bus bit-45
mem_sdr_data_b_44			BI			Controller B data bus bit-44
mem_sdr_data_b_43			BI			Controller B data bus bit-43
mem_sdr_data_b_49			BI			Controller B data bus bit-42

mem_sdr_data_b_41			BI			Controller B data bus bit-41
mem_sdr_data_b_40			BI			Controller B data bus bit-40
mem_sdr_data_b_39			BI			Controller B data bus bit-39
mem_sdr_data_b_38			BI			Controller B data bus bit-38
mem_sdr_data_b_37			BI			Controller B data bus bit-37
mem_sdr_data_b_36			BI			Controller B data bus bit-36
mem_sdr_data_b_35			BI			Controller B data bus bit-35
mem_sdr_data_b_34			BI			Controller B data bus bit-34
mem_sdr_data_b_33			BI			Controller B data bus bit-33
mem_sdr_data_b_32			BI			Controller B data bus bit-32
mem_sdr_data_b_31			BI			Controller B data bus bit-31
mem_sdr_data_b_30			BI			Controller B data bus bit-30
mem_sdr_data_b_29			BI			Controller B data bus bit-29
mem_sdr_data_b_28			BI			Controller B data bus bit-28
mem_sdr_data_b_27			BI			Controller B data bus bit-27
mem_sdr_data_b_26			BI			Controller B data bus bit-26
mem_sdr_data_b_25			BI			Controller B data bus bit-25
mem_sdr_data_b_24			BI			Controller B data bus bit-24
mem_sdr_data_b_23			BI			Controller B data bus bit-23
mem_sdr_data_b_22			BI			Controller B data bus bit-22
mem_sdr_data_b_21			BI			Controller B data bus bit-22
mem_sdr_data_b_20			BI			Controller B data bus bit-20
mem_sdr_data_b_19			BI			Controller B data bus bit-19

mem_sdr_data_b_18			BI			Controller B data bus bit-18
mem_sdr_data_b_17			BI			Controller B data bus bit-17
mem_sdr_data_b_16			BI			Controller B data bus bit-16
mem_sdr_data_b_15			BI			Controller B data bus bit-15
mem_sdr_data_b_14			BI			Controller B data bus bit-14
mem_sdr_data_b_13			BI			Controller B data bus bit-13
mem_sdr_data_b_12			BI			Controller B data bus bit-12
mem_sdr_data_b_11			BI			Controller B data bus bit-11
mem_sdr_data_b_10			BI			Controller B data bus bit-10
mem_sdr_data_b_9			BI			Controller B data bus bit-9
mem_sdr_data_b_8			BI			Controller B data bus bit-8
mem_sdr_data_b_7			BI			Controller B data bus bit-7
mem_sdr_data_b_6			BI			Controller B data bus bit-6
mem_sdr_data_b_5			BI			Controller B data bus bit-5
mem_sdr_data_b_4			BI			Controller B data bus bit-4
mem_sdr_data_b_3			BI			Controller B data bus bit-3
mem_sdr_data_b_2			BI			Controller B data bus bit-2
mem_sdr_data_b_1			BI			Controller B data bus bit-1
mem_sdr_data_b_0			BI			Controller B data bus bit-0
mem_sdr_addr0_b_11			OUT			Controller B addr0 bit-11
mem_sdr_addr0_b_10			OUT			Controller B addr0 bit-10
mem_sdr_addr0_b_9			OUT			Controller B addr0 bit-9
mem_sdr_addr0_b_8			OUT			Controller B addr0 bit-8

mem_sdr_addr0_b_7			OUT			Controller B addr0 bit-7
mem_sdr_addr0_b_6			OUT			Controller B addr0 bit-6
mem_sdr_addr0_b_5			OUT			Controller B addr0 bit-5
mem_sdr_addr0_b_4			OUT			Controller B addr0 bit-4
mem_sdr_addr0_b_3			OUT			Controller B addr0 bit-3
mem_sdr_addr0_b_2			OUT			Controller B addr0 bit-2
mem_sdr_addr0_b_1			OUT			Controller B addr0 bit-1
mem_sdr_addr0_b_0			OUT			Controller B addr0 bit-0
mem_sdr_addr1_b_11			OUT			Controller B addr1 bit-11
mem_sdr_addr1_b_10			OUT			Controller B addr1 bit-10
mem_sdr_addr1_b_9			OUT			Controller B addr1 bit-9
mem_sdr_addr1_b_8			OUT			Controller B addr1 bit-8
mem_sdr_addr1_b_7			OUT			Controller B addr1 bit-7
mem_sdr_addr1_b_6			OUT			Controller B addr1 bit-6
mem_sdr_addr1_b_5			OUT			Controller B addr1 bit-5
mem_sdr_addr1_b_4			OUT			Controller B addr1 bit-4
mem_sdr_addr1_b_3			OUT			Controller B addr1 bit-3
mem_sdr_addr1_b_2			OUT			Controller B addr1 bit-2
mem_sdr_addr1_b_1			OUT			Controller B addr1 bit-1
mem_sdr_addr1_b_0			OUT			Controller B addr1 bit-0
mem_sdr_ bank0_b_1			OUT			Controller B bank0 select bit-1
mem_sdr_ bank0_b_0			OUT			Controller B bank0 select bit-0
mem_sdr_ bank1_b_1			OUT			Controller B bank1 select bit-1

mem_sdr_ bank1_b_0			OUT			Controller B bank1 select bit-0
mem_sdr_cs_b_7			OUT			Controller B chip select bit-7
mem_sdr_cs_b_6			OUT			Controller B chip select bit-6
mem_sdr_cs_b_5			OUT			Controller B chip select bit-5
mem_sdr_cs_b_4			OUT			Controller B chip select bit-4
mem_sdr_cs_b_3			OUT			Controller B chip select bit-3
mem_sdr_cs_b_2			OUT			Controller B chip select bit-2
mem_sdr_cs_b_1			OUT			Controller B chip select bit-1
mem_sdr_cs_b_0			OUT			Controller B chip select bit-0
mem_sdr_dqs0_b_1			BI			Controller B data strobe0 bit-1
mem_sdr_dqs0_b_0			BI			Controller B data strobe0 bit-0
mem_sdr_dqs1_b_1			BI			Controller B data strobe1 bit-1
mem_sdr_dqs1_b_0			BI			Controller B data strobe1 bit-0
mem_sdr_ras0_b			OUT			Controller B RAS0
mem_sdr_cas0_b			OUT			Controller B CAS0
mem_sdr_we0_b			OUT			Controller B write enable0
mem_sdr_clk0_b			OUT			Controller B differential clock output0
mem_sdr_clk0_b_1			OUT			Controller B differential clock output0, low active
mem_sdr_ras1_b			OUT			Controller B RAS1
mem_sdr_cas1_b			OUT			Controller B CAS1
mem_sdr_we1_b			OUT			Controller B write enable1
mem_sdr_clk1_b			OUT			Controller B differential clock output1
mem_sdr_clk1b_1			OUT			Controller B differential clock output1, low active

mem_sdr_clke_b			OUT			Controller B clock enable
Controller B = 115 pins						
Memory = 228 pins with 2 controllers						
pfq-mem_clk			IN			
pfq-mem_valid			IN			
pfq-mem_com_1			IN			
pfq-mem_com_0			IN			
pfq_mem_data_23			IN			PFQ=>MEM data bit-23
pfq_mem_data_22			IN			PFQ=>MEM data bit-22
pfq_mem_data_21			IN			PFQ=>MEM data bit-21
pfq_mem_data_20			IN			PFQ=>MEM data bit-20
pfq_mem_data_19			IN			PFQ=>MEM data bit-19
pfq_mem_data_18			IN			PFQ=>MEM data bit-18
pfq_mem_data_17			IN			PFQ=>MEM data bit-17
pfq_mem_data_16			IN			PFQ=>MEM data bit-16
pfq_mem_data_15			IN			PFQ=>MEM data bit-15
pfq_mem_data_14			IN			PFQ=>MEM data bit-14
pfq_mem_data_13			IN			PFQ=>MEM data bit-13
pfq_mem_data_12			IN			PFQ=>MEM data bit-12
pfq_mem_data_11			IN			PFQ=>MEM data bit-11
pfq_mem_data_10			IN			PFQ=>MEM data bit-10
pfq_mem_data_9			IN			PFQ=>MEM data bit-9
pfq_mem_data_8			IN			PFQ=>MEM data bit-8
pfq_mem_data_7			IN			PFQ=>MEM data bit-7
pfq_mem_data_6			IN			PFQ=>MEM data bit-6
pfq_mem_data_5			IN			PFQ=>MEM data bit-5
pfq_mem_data_4			IN			PFQ=>MEM data bit-4
pfq_mem_data_3			IN			PFQ=>MEM data bit-3
pfq_mem_data_2			IN			PFQ=>MEM data bit-2

pfq_mem_data_1			IN			PFQ=>MEM data bit-1
pfq_mem_data_0			IN			PFQ=>MEM data bit-0
mem_pfq_data_19			OUT			MEM=>PFQ data bit-19
mem_pfq_data_18			OUT			MEM=>PFQ data bit-18
mem_pfq_data_17			OUT			MEM=>PFQ data bit-17
mem_pfq_data_16			OUT			MEM=>PFQ data bit-16
mem_pfq_data_15			OUT			MEM=>PFQ data bit-15
mem_pfq_data_14			OUT			MEM=>PFQ data bit-14
mem_pfq_data_13			OUT			MEM=>PFQ data bit-13
mem_pfq_data_12			OUT			MEM=>PFQ data bit-12
mem_pfq_data_11			OUT			MEM=>PFQ data bit-11
mem_pfq_data_10			OUT			MEM=>PFQ data bit-10
mem_pfq_data_9			OUT			MEM=>PFQ data bit-9
mem_pfq_data_8			OUT			MEM=>PFQ data bit-8
mem_pfq_data_7			OUT			MEM=>PFQ data bit-7
mem_pfq_data_6			OUT			MEM=>PFQ data bit-6
mem_pfq_data_5			OUT			MEM=>PFQ data bit-5
mem_pfq_data_4			OUT			MEM=>PFQ data bit-4
mem_pfq_data_3			OUT			MEM=>PFQ data bit-3
mem_pfq_data_2			OUT			MEM=>PFQ data bit-2
mem_pfq_data_1			OUT			MEM=>PFQ data bit-1
mem_pfq_data_0			OUT			MEM=>PFQ data bit-0
mem_pfq_clk			OUT			MEM=>PFQ clock
mem_pfq_valid			OUT			MEM=>PFQ valid signal
mem_pfq_com_1			OUT			MEM=>PFQ command bit-1
mem_pfq_com_0			OUT			MEM=>PFQ command bit-0
mem_pfq_full			OUT			MEM=>PFQ full
To PFQ = 53 pins						
Total pins = 223 pins or 308 pins						
Reassembly Engine						
Pin Name	**Pad#**	**Pin#**	**Dir**	**Input Buffer**	**Output Buffer**	**Description**

ras_mem_i_clk			IN			Memory clock input to reassembly
ras_mem_o_clk			OUT			Memory clock output from reassembly
ras_mem_we_l			OUT			Memory write enable, active low
ras_mem_cs_l_7			OUT			Memory chip select bit-7, active low
ras_mem_cs_l_6			OUT			Memory chip select bit-6, active low
ras_mem_cs_l_5			OUT			Memory chip select bit-5, active low
ras_mem_cs_l_4			OUT			Memory chip select bit-4, active low
ras_mem_cs_l_3			OUT			Memory chip select bit-3, active low
ras_mem_cs_l_2			OUT			Memory chip select bit-2, active low
ras_mem_cs_l_1			OUT			Memory chip select bit-1, active low
ras_mem_cs_l_0			OUT			Memory chip select bit-0, active low
ras_mem_addr_21			OUT			Memory address bit-21
ras_mem_addr_20			OUT			Memory address bit-20
ras_mem_addr_19			OUT			Memory address bit-19
ras_mem_addr_18			OUT			Memory address bit-18
ras_mem_addr_17			OUT			Memory address bit-17
ras_mem_addr_16			OUT			Memory address bit-16
ras_mem_addr_15			OUT			Memory address bit-15
ras_mem_addr_14			OUT			Memory address bit-14
ras_mem_addr_13			OUT			Memory address bit-13
ras_mem_addr_12			OUT			Memory address bit-12
ras_mem_addr_11			OUT			Memory address bit-11
ras_mem_addr_10			OUT			Memory address bit-10
ras_mem_addr_9			OUT			Memory address bit-9
ras_mem_addr_8			OUT			Memory address bit-8
ras_mem_addr_7			OUT			Memory address bit-7
ras_mem_addr_6			OUT			Memory address bit-6
ras_mem_addr_5			OUT			Memory address bit-5
ras_mem_addr_4			OUT			Memory address bit-4
ras_mem_addr_3			OUT			Memory address bit-3
ras_mem_addr_2			OUT			Memory address bit-2
ras_mem_addr_1			OUT			Memory address bit-1
ras_mem_addr_0			OUT			Memory address bit-0
ras_mem_data_35			BI			Memory data bit-35
ras_mem_data_34			BI			Memory data bit-34

ras_mem_data_33			BI			Memory data bit-33
ras_mem_data_32			BI			Memory data bit-32
ras_mem_data_31			BI			Memory data bit-31
ras_mem_data_30			BI			Memory data bit-30
ras_mem_data_29			BI			Memory data bit-29
ras_mem_data_28			BI			Memory data bit-28
ras_mem_data_27			BI			Memory data bit-27
ras_mem_data_26			BI			Memory data bit-26
ras_mem_data_25			BI			Memory data bit-25
ras_mem_data_24			BI			Memory data bit-24
ras_mem_data_23			BI			Memory data bit-23
ras_mem_data_22			BI			Memory data bit-22
ras_mem_data_21			BI			Memory data bit-21
ras_mem_data_20			BI			Memory data bit-20
ras_mem_data_19			BI			Memory data bit-19
ras_mem_data_18			BI			Memory data bit-18
ras_mem_data_17			BI			Memory data bit-17
ras_mem_data_16			BI			Memory data bit-16
ras_mem_data_15			BI			Memory data bit-15
ras_mem_data_14			BI			Memory data bit-14
ras_mem_data_13			BI			Memory data bit-13
ras_mem_data_12			BI			Memory data bit-12
ras_mem_data_11			BI			Memory data bit-11
ras_mem_data_10			BI			Memory data bit-10
ras_mem_data_9			BI			Memory data bit-9
ras_mem_data_8			BI			Memory data bit-8
ras_mem_data_7			BI			Memory data bit-7
ras_mem_data_6			BI			Memory data bit-6
ras_mem_data_5			BI			Memory data bit-5
ras_mem_data_4			BI			Memory data bit-4
ras_mem_data_3			BI			Memory data bit-3
ras_mem_data_2			BI			Memory data bit-2
ras_mem_data_1			BI			Memory data bit-1
ras_mem_data_0			BI			Memory data bit-0
Memory = 69 pins						
Total pins = 69						
CPU Interface						
Pin Name	**Pad#**	**Pin#**	**Dir**	**Input Buffer**	**Output Buffer**	Description
cpu_config_1						Address/data multiplex select 0: Non-multiplexed 1: Multiplexed
cpu_config_0						Endian select 0: Big endian

						1: Little endian
cpu_cs_l						CPU chip select, active low
cpu_rdwr_l						CPU read/write strobe
cpu_addr_9						CPU address bit-9
cpu_addr_8						CPU address bit-8
cpu_addr_7						CPU address bit-7
cpu_addr_6						CPU address bit-6
cpu_addr_5						CPU address bit-5
cpu_addr_4						CPU address bit-4
cpu_addr_3						CPU address bit-3
cpu_addr_2						CPU address bit-2
cpu_addr_1						CPU address bit-1
cpu_addr_0						CPU address bit-0
cpu_data_31						CPU data bit-31
cpu_data_30						CPU data bit-30
cpu_data_29						CPU data bit-29
cpu_data_28						CPU data bit-28
cpu_data_27						CPU data bit-27
cpu_data_26						CPU data bit-26
cpu_data_25						CPU data bit-25
cpu_data_24						CPU data bit-24
cpu_data_23						CPU data bit-23
cpu_data_22						CPU data bit-22
cpu_data_21						CPU data bit-21
cpu_data_20						CPU data bit-20
cpu_data_19						CPU data bit-19
cpu_data_18						CPU data bit-18
cpu_data_17						CPU data bit-17
cpu_data_16						CPU data bit-16
cpu_data_15						CPU data bit-15
cpu_data_14						CPU data bit-14
cpu_data_13						CPU data bit-13
cpu_data_12						CPU data bit-12
cpu_data_11						CPU data bit-11
cpu_data_10						CPU data bit-10
cpu_data_9						CPU data bit-9
cpu_data_8						CPU data bit-8
cpu_data_7						CPU data bit-7
cpu_data_6						CPU data bit-6
cpu_data_5						CPU data bit-5
cpu_data_4						CPU data bit-4
cpu_data_3						CPU data bit-3
cpu_data_2						CPU data bit-2
cpu_data_1						CPU data bit-1

cpu_data_0						CPU data bit-0
Total pins = 46						
MISC						
Pin Name	**Pad#**	**Pin#**	**Dir**	**Input Buffer**	**Output Buffer**	**Description**
test_mux_31			OUT			Test mux out bit-31
test_mux_30			OUT			Test mux out bit-30
test_mux_29			OUT			Test mux out bit-29
test_mux_28			OUT			Test mux out bit-28
test_mux_27			OUT			Test mux out bit-27
test_mux_26			OUT			Test mux out bit-26
test_mux_25			OUT			Test mux out bit-25
test_mux_24			OUT			Test mux out bit-24
test_mux_23			OUT			Test mux out bit-23
test_mux_22			OUT			Test mux out bit-22
test_mux_21			OUT			Test mux out bit-21
test_mux_20			OUT			Test mux out bit-20
test_mux_19			OUT			Test mux out bit-19
test_mux_18			OUT			Test mux out bit-18
test_mux_17			OUT			Test mux out bit-17
test_mux_16			OUT			Test mux out bit-16
test_mux_15			OUT			Test mux out bit-15
test_mux_14			OUT			Test mux out bit-14
test_mux_13			OUT			Test mux out bit-13
test_mux_12			OUT			Test mux out bit-12
test_mux_11			OUT			Test mux out bit-11
test_mux_10			OUT			Test mux out bit-10
test_mux_9			OUT			Test mux out bit-9
test_mux_8			OUT			Test mux out bit-8
test_mux_7			OUT			Test mux out bit-7
test_mux_6			OUT			Test mux out bit-6
test_mux_5			OUT			Test mux out bit-5
test_mux_4			OUT			Test mux out bit-4
test_mux_3			OUT			Test mux out bit-3
test_mux_2			OUT			Test mux out bit-2
test_mux_1			OUT			Test mux out bit-1
test_mux_0			OUT			Test mux out bit-0
testmux_clkout			OUT			Test mux clkout
Tck			IN			JTAG clock
Tms			IN			JTAG mux select
tdi_scan_in			IN			JTAG data in, or scan in
tdo_bist_scan_out			OUT			JTAG data out, or scan out
trst_l			IN			JTAG reset, active low

Total pins = 38
Grand total = 723 with 2 DDRs

Figure 22

#	Block's name	Address [9:6]	Start Address	Last Address	Total length
1	CPU Interface	0000b	0000000b	1111111b	64d
2	Incoming SPI-4	0001b			64d
3	Lookup Engine	0010b			64d
4	Segmentation	0011b			64d
5	Memory Manager	0100b			64d
6	Reassembly	0101b			64d
7	Outgoing SPI-4	0110b			64d
8 : 16	Reserved	0111b - 1111b			64d : 64d

Figure 23

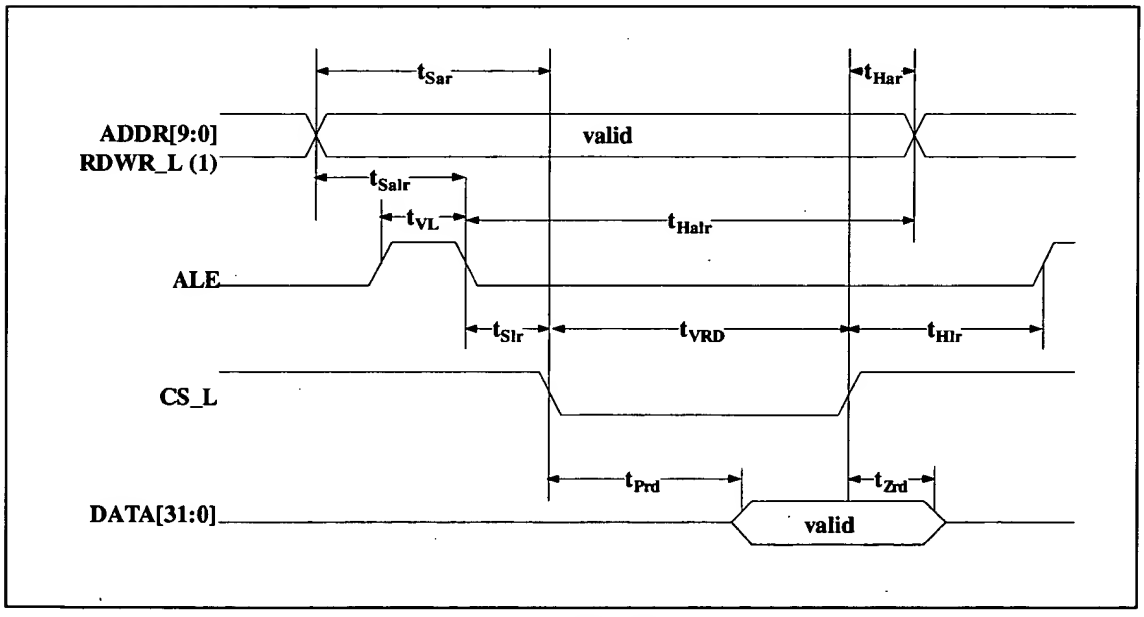


Figure 24

Symbol	Parameter	Min	Max	Units
t_{Sar}	Address/rdwr_l to Valid Read set-up time	0		ns
t_{Har}	Address/rdwr_l to Valid read hold time	0		ns
t_{Salr}	Address to latch set-up time	5		ns
T_{Halr}	Address to latch hold time	5		ns
T_{VL}	Valid latch pulse width	5		ns
T_{Slr}	Latch to Read set-up	0		ns
T_{Hlr}	Latch to Read hold	5		ns
T_{Prd}	Valid Read to valid data propagation delay		50	ns
t_{Zrd}	Valid Read negated to output tri-state		10	ns
t_{VRD}	Valid Read pulse width	60		

Figure 25

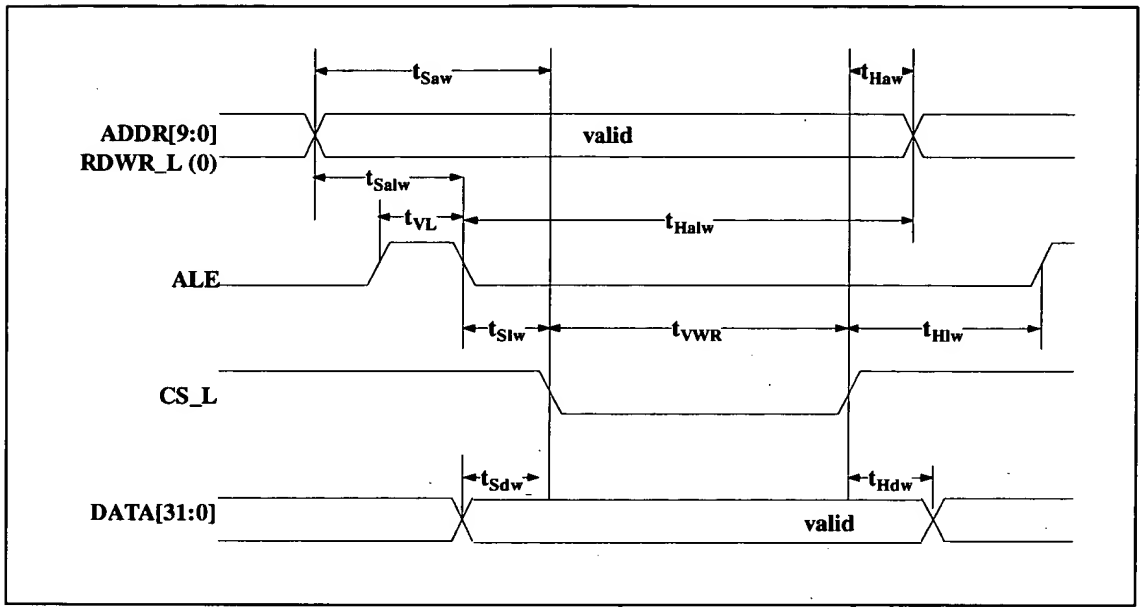


Figure 26

Symbol	Parameter	Min	Max	Units
t_{Saw}	Address/rdwr_l to Valid Write set-up time	0		ns
t_{Haw}	Address/rdwr_l to Valid Write hold time	0		ns
t_{Salw}	Address to latch set-up time	5		ns
T_{Halw}	Address to latch hold time	5		ns
T_{VL}	Valid latch pulse width	5		ns
T_{Slw}	Latch to Write set-up	0		ns
T_{Hlw}	Latch to Write hold	5		ns
T_{Sdw}	Data to valid Write set-up time	0		ns
T_{Hdw}	Data to valid write hold time	5		ns
T_{VWR}	Valid Write pulse width	60		ns

Figure 27

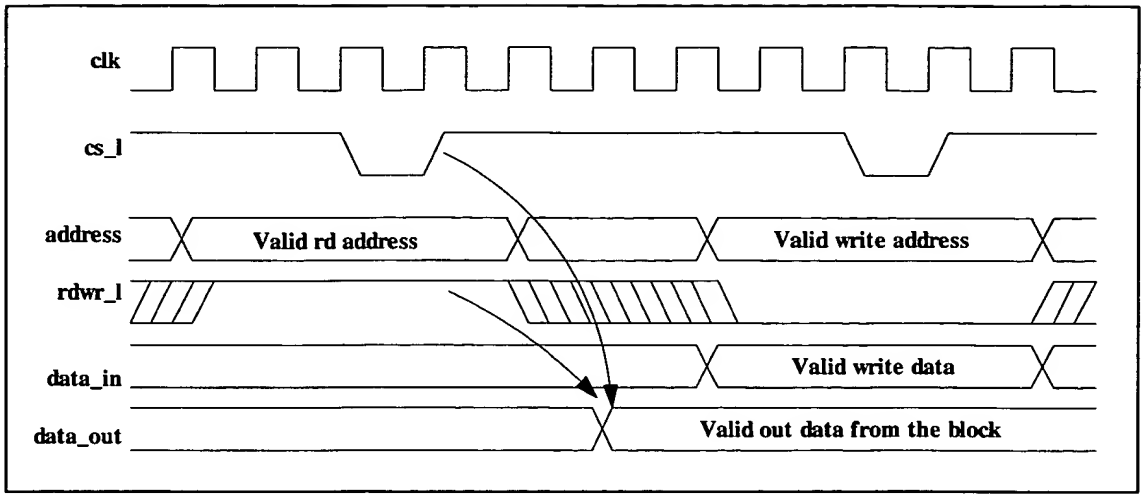


Figure 28

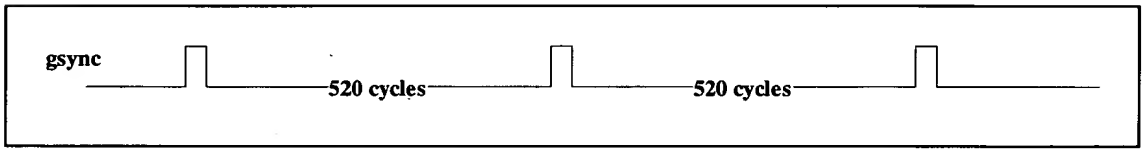


Figure 29

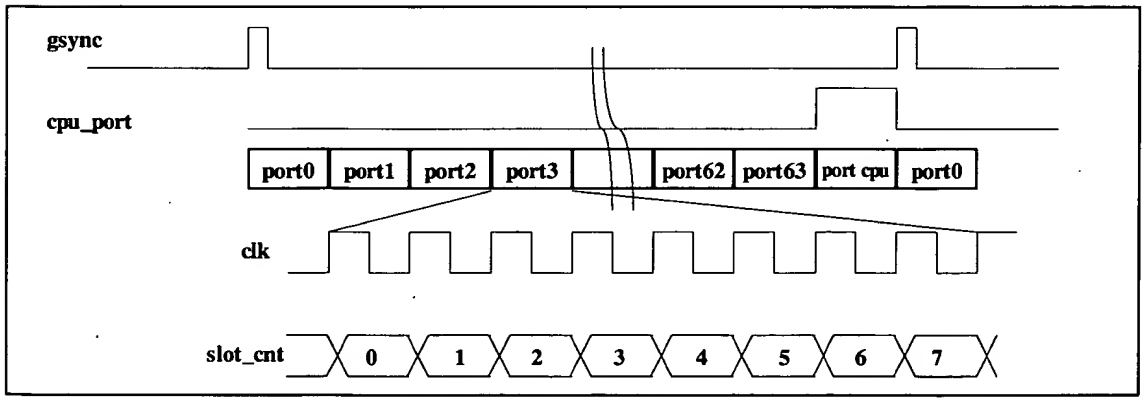


Figure 30

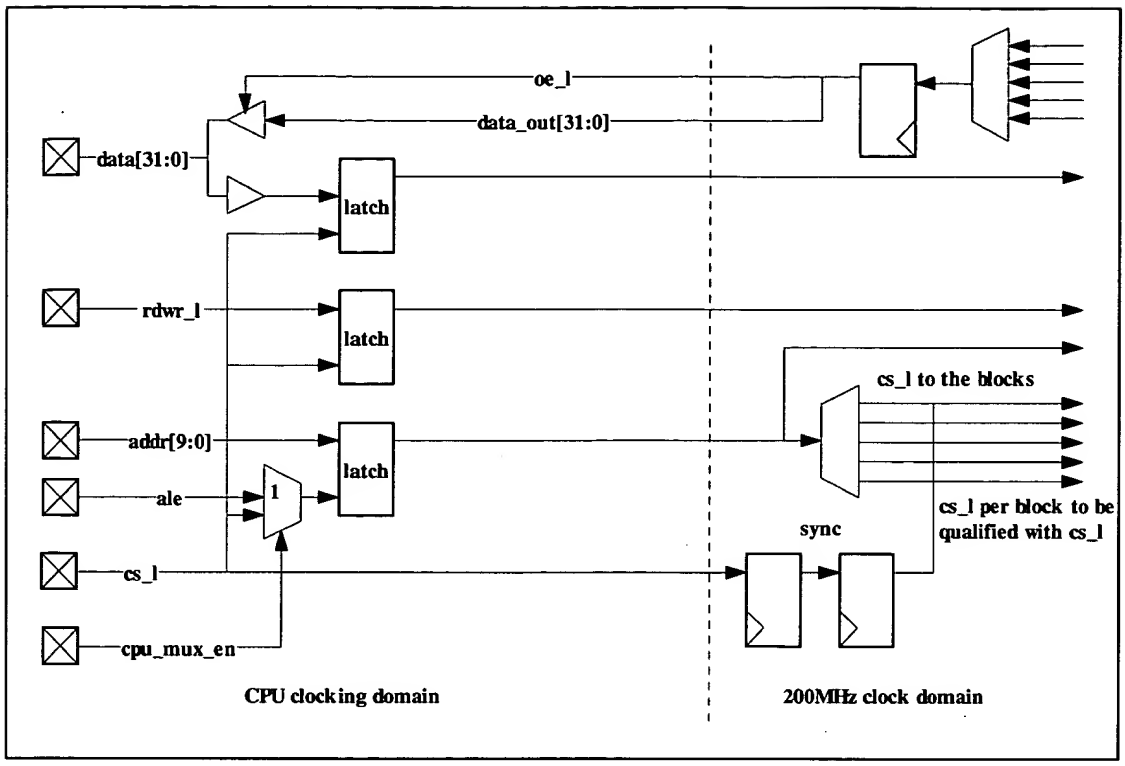

oe_l
data_out[31:0]
data[31:0]
latch
rdwr_l
latch
cs_l to the blocks
addr[9:0]
latch
ale
1
sync
cs_l per block to be qualified with cs_l
cs_l
cpu_mux_en
CPU clocking domain
200MHz clock domain

Figure 31.

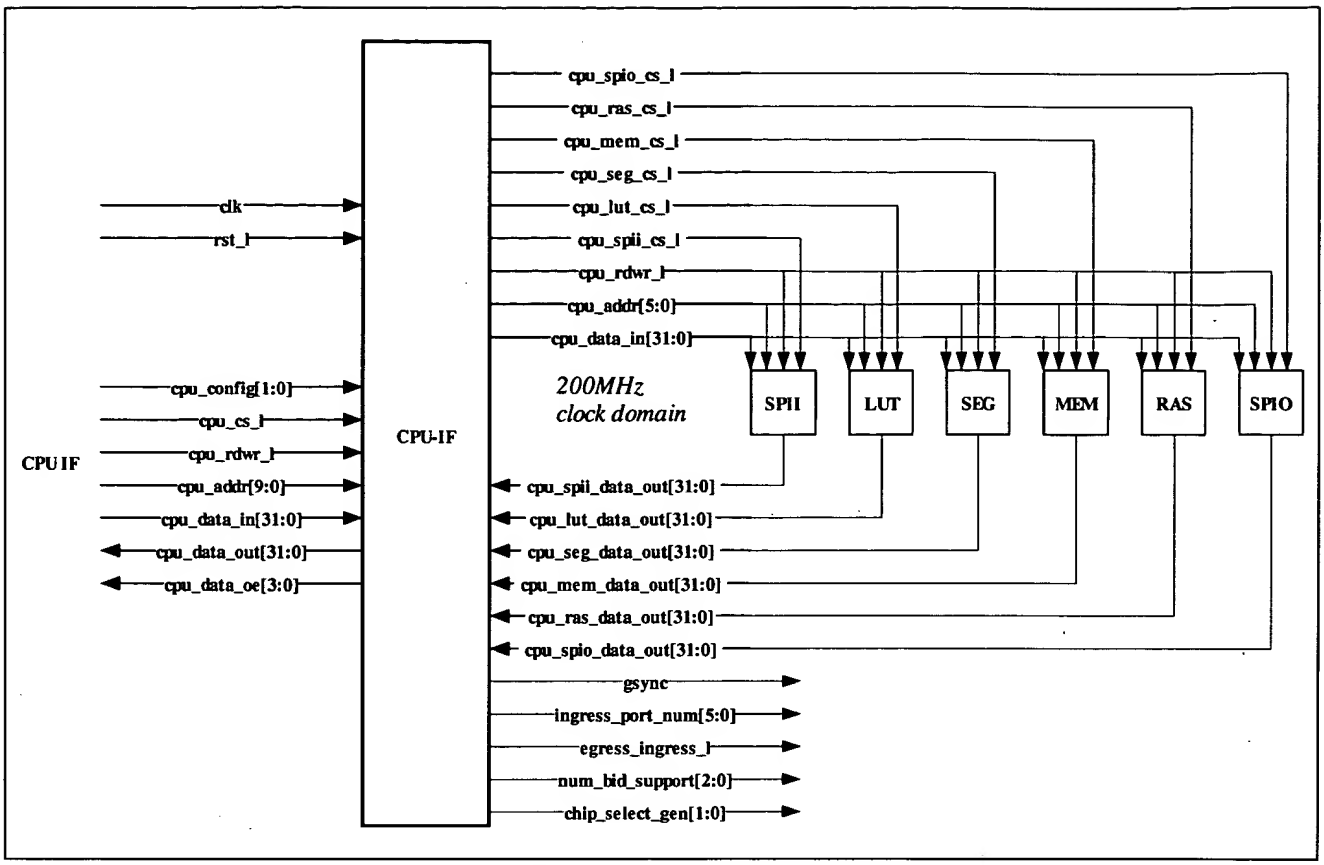

CPU IF
clk
rst_l
cpu_config[1:0]
cpu_cs_l
cpu_rdwr_l
cpu_addr[9:0]
cpu_data_in[31:0]
cpu_data_out[31:0]
cpu_data_oe[3:0]
CPU-IF
cpu_spio_cs_l
cpu_ras_cs_l
cpu_mem_cs_l
cpu_seg_cs_l
cpu_lut_cs_l
cpu_spii_cs_l
cpu_rdwr_l
cpu_addr[5:0]
cpu_data_in[31:0]
200MHz
clock domain
SPII
LUT
SEG
MEM
RAS
SPIO
cpu_spii_data_out[31:0]
cpu_lut_data_out[31:0]
cpu_seg_data_out[31:0]
cpu_mem_data_out[31:0]
cpu_ras_data_out[31:0]
cpu_spio_data_out[31:0]
gsync
ingress_port_num[5:0]
egress_ingress_l
num_bid_support[2:0]
chip_select_gen[1:0]

Figure 32

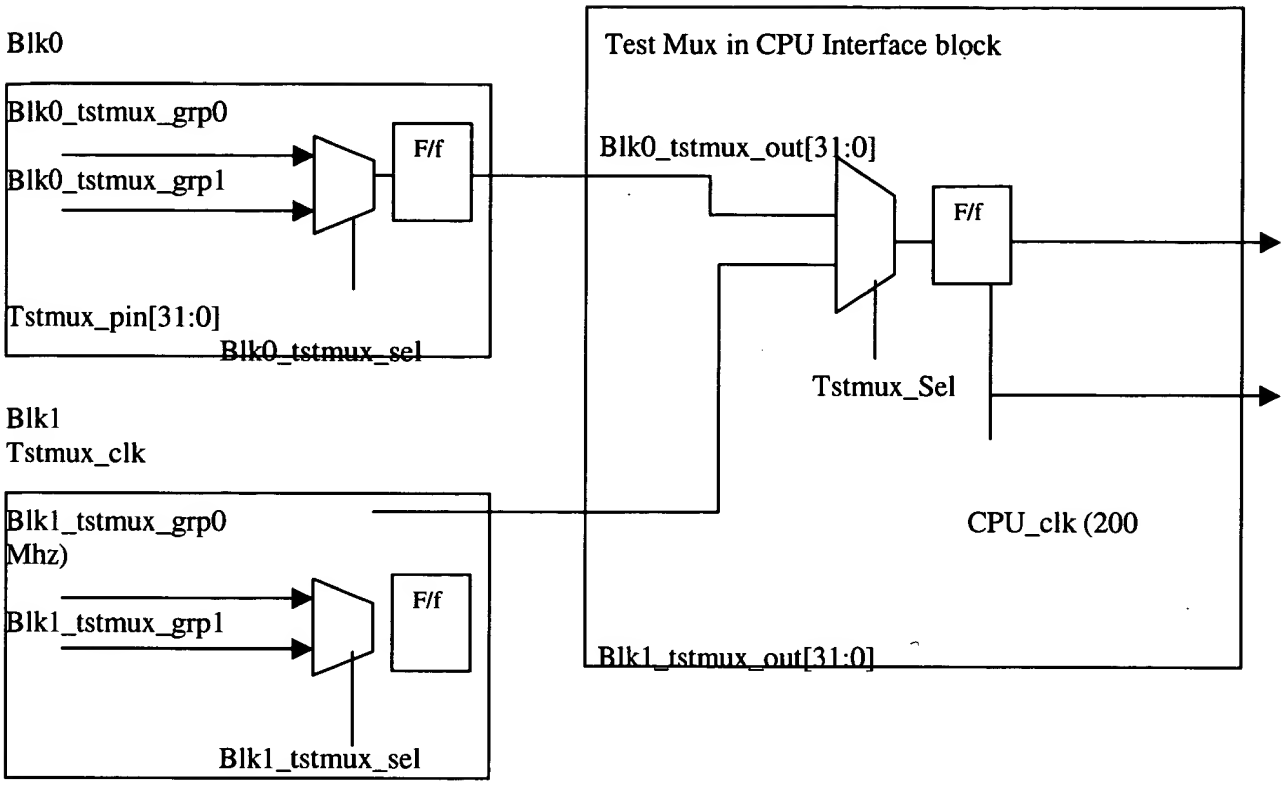


Figure 33

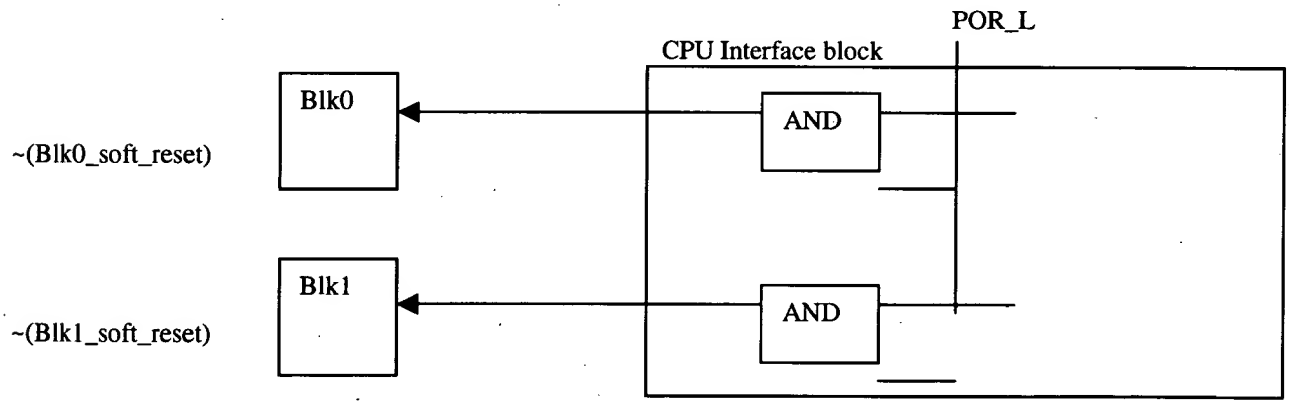


Figure 34

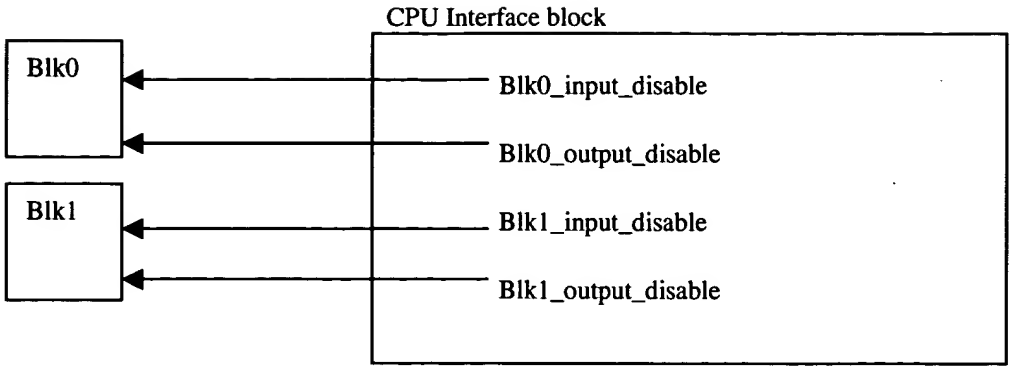


Figure 35

Address	Name	Type	Description
0 – 31	Reserved		
32	VERSION	R	Read only register to contains the version of the chip. It's value is 0000_0001h
33	MOD_CTRL	RW	[5:0] – Port number for ingress chip Default value: 0 [7:6] – Reserved [8] – Ingress/Egress chip selection 0 – ingress chip 1 – egress chip Default value: 1
34	MEM_CTRL	RW	[2:0] – Number of supported BIDs 000 – 1M BIDs 001 – 2M BIDs 010 – 3M BIDs 011 – 4M BIDs 100 – 5M BIDs 101 – 6M BIDs 110 – 7M BIDs 111 – 8M BIDs Default value: 001 [3] – Reserved [5:4] – Chip select generation 00 – based on address[20:19] 01 - based on address[21:20] 10 - based on address[22:21] 11 – Reserved Default value: 00 [31:6] – Reserved

35	Soft_Reset	RW	[0] – Soft reset register for Incoming SPI-4 block. 0: normal operation (default) 1: reset block [1] – Soft reset register for Outgoing SPI-4 block. 0: normal operation (default) 1: reset block [2] – Soft reset register for Lookup Engine block. 0: normal operation (default) 1: reset block [3] – Soft reset register for Segmentation block. 0: normal operation (default) 1: reset block [4] – Soft reset register for Memory Manager block. 0: normal operation (default) 1: reset block [5] – Soft reset register for Reassembly block. 0: normal operation (default) 1: reset block [31:6] - Reserved

36	Input/Output_Disable	RW	[0] – Input Disable for Incoming SPI-4 0: enable (default) 1: disable [1] – Output Disable for Incoming SPI-4 0: enable (default) 1: disable [2] – Input Disable for Outgoing SPI-4 0: enable (default) 1: disable [3] – Output Disable for Outgoing SPI-4 0: enable (default) 1: disable [4] – Input Disable for Lookup Engine 0: enable (default) 1: disable [5] – Output Disable for Lookup Engine 0: enable (default) 1: disable [6] – Input Disable for Segmentation 0: enable (default) 1: disable [7] – Output Disable for Segmentation 0: enable (default) 1: disable [8] – Input Disable for Memory Manager 0: enable (default) 1: disable [9] – Output Disable for Memory Manager 0: enable (default) 1: disable [10] – Input Disable for Reassembly 0: enable (default) 1: disable [11] – Output Disable for Reassembly 0: enable (default) 1: disable [31:12] – Reserved

37	TSTMUX_SEL	RW	[3:0] – Block selector for Test Mux 0000: No output (default) 0001: Incoming SPI-4 0010: Outgoing SPI-4 0011: Lookup Engine 0100: Segmentation 0101: Memory Manager 0110: Reassembly Others: No output [31:4] - Reserved
38-63	Reserved		

Figure 36

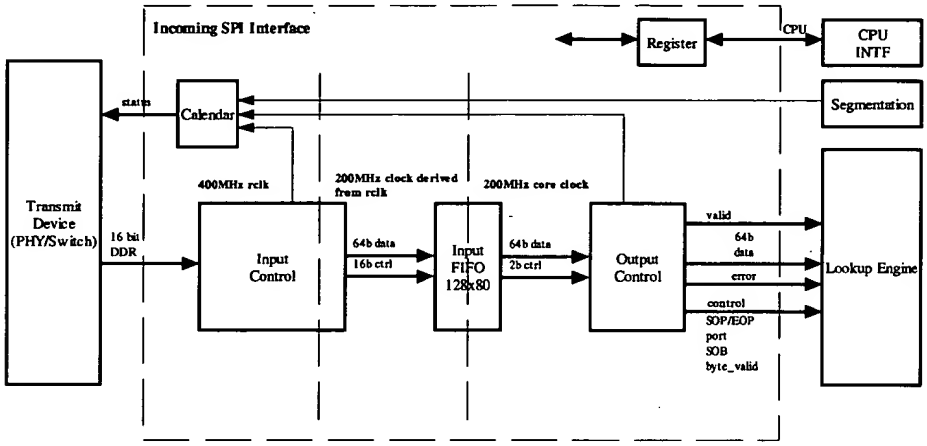


Figure 37

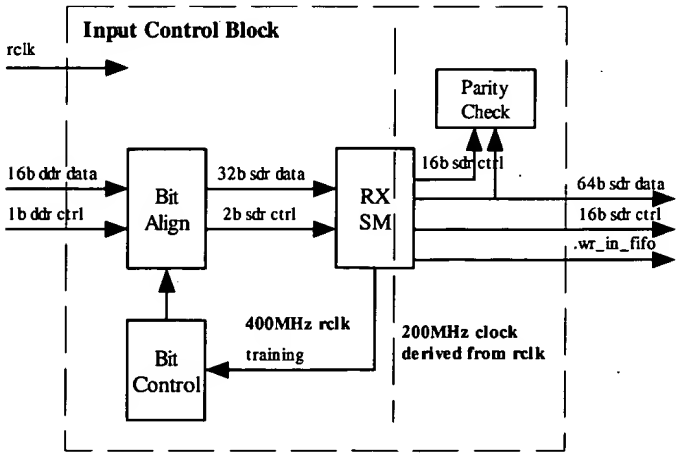


Figure 38

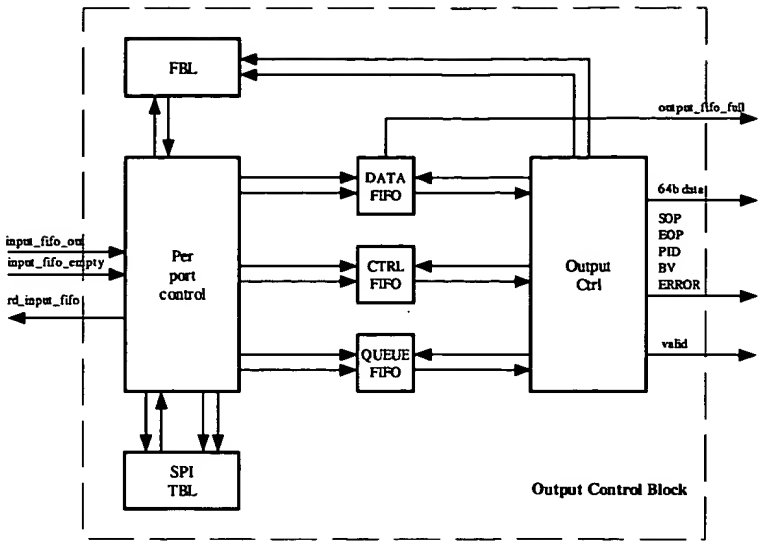


Figure 39

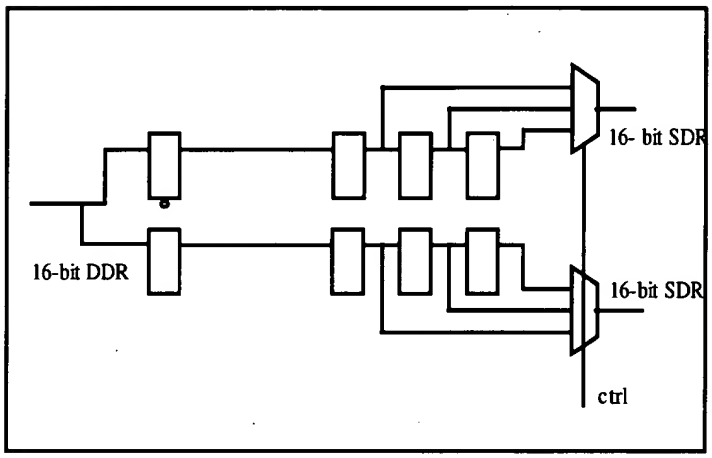


Figure 40

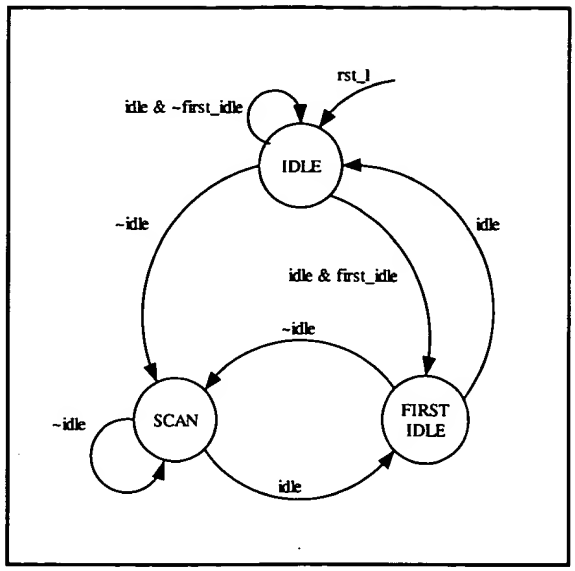


Figure 41

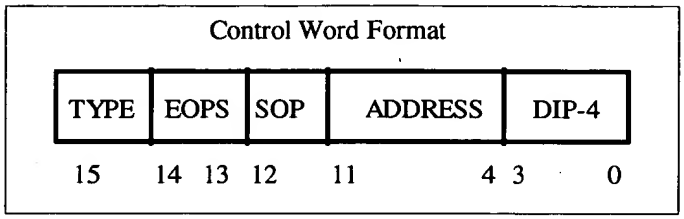


Figure 42

Bit Position	Label	Description
15	Type	Control Word Type. Set to either of the following values: 1: payload control word (payload transfer will immediately follow the control word). 0: idle or training control word (otherwise).
14:13	EOPS	End-of-Packet (EOP) Status. Set to the following values below according to the status of the immediately preceding payload transfer. 0 0: Not an EOP. 0 1: EOP Abort (application-specific error condition). 1 0: EOP Normal termination, 2 bytes valid. 1 1: EOP Normal termination, 1 byte valid.

Bit Position	Label	Description
		EOPS is valid in the first control word following a burst transfer. It is ignored and set to "0 0" otherwise.
12	SOP	Start-of-Packet. Set to 1 if the payload transfer immediately following the control word corresponds to the start of a packet. Set to 0 otherwise. Set to 0 in all idle and training control words.
11:4	ADR	Port Address. 8-bit port address of the payload data transfer immediately following the control word. None of the addresses are reserved (all are available for payload transfer). Set to all zeroes in all idle control words. Set to all ones in all training control words.
3:0	DIP-4	4-bit Diagonal Interleaved Parity. 4-bit odd parity computed over the current control word and the immediately preceding data words (if any) following the last control word.

Figure 43

	Bit [15:12]	Next Word Status	Prior Word Status	Meaning
0	0000	Idle	Continued	Idle, not EOP, training control word
1	0001	Reserved	Reserved	Reserved
2	0010	Idle	EOP w/abort	Idle, Abort last packet
3	0011	Reserved	Reserved	Reserved
4	0100	Idle	EOP w/1 byte	Idle, EOP with 2 bytes valid
5	0101	Reserved	Reserved	Reserved
6	0110	Idle	EOP w/2 bytes	Idle, EOP with 1 byte valid
7	0111	Reserved	Reserved	Reserved
8	1000	Valid	None	Valid, no SOP, no EOP
9	1001	Valid/SOP	None	Valid, SOP, no EOP
A	1010	Valid	EOP w/abort	Valid, no SOP, abort

	Bit [15:12]	Next Word Status	Prior Word Status	Meaning
B	1011	Valid/SOP	EOP w/abort	Valid, SOP, abort
C	1100	Valid	EOP w/ 2 bytes	Valid, no SOP, EOP with 2 bytes valid
D	1101	Valid	EOP w/ 2 bytes	Valid, SOP, EOP with 2 bytes valid
E	1110	Valid	EOP w/1 byte	Valid, no SOP, EOP with 1 byte valid
F	1111	Valid	EOP w/1 byte	Valid, SOP, EOP with 1 byte valid

Figure 44

Current State	CTRL	Word 1	Word 2	Next State		Comments
IDLE	11	I_ctrl	I_ctrl	IDLE		
	11	I_ctrl	P_ctrl	PLOAD		
	11	I_ctrl	T_ctrl	TRAIN_C		
	11	T_ctrl	T_ctrl	TRAIN_C		
	10	P_ctrl	Data	DATA1		
	01/ 00					Error
TRAIN_C	11	T_ctrl	T_ctrl	TRAIN_C		
	10	T_ctrl	T_data	TRAIN_D		
	00	T_data	T_data	TRAIN_D		
	01					Error
TRAIN_D	00	T_data	T_data	TRAIN_D		
	01	T_data	P_ctrl	PLOAD		
	01	T_data	I_ctrl	IDLE		
	01	T_data	T_ctrl	TRAIN_C		
	10/ 11					Error
PLOAD	00	Data	Data	DATA2		
	01	Data	P_ctrl	PLOAD		
	01	Data	I_ctrl	IDLE		
	10/ 11					Error

DATA1	00	Data	Data	DATA3		
	01	Data	P_ctrl	PLOAD		
	01	Data	I_ctrl	IDLE		
	10	P_ctrl	data	DATA1		
	11	I_ctrl	I_ctrl	IDLE		
	11	I-ctrl	P_ctrl	PLOAD		
DATA2	00	data	data	DATA4		
	01	data	P_ctrl	PLOAD		
	01	data	I_ctrl	IDLE		
	10	P_ctrl	data	DATA1		
	11	I_ctrl	I_ctrl	IDLE		
	11	I_ctrl	P_ctrl	PLOAD		
DATA3	00	data	data	DATA5		
	01	data	P_ctrl	PLOAD		
	01	data	I_ctrl	IDLE		
	10	P_ctrl	data	DATA1		
	11	I_ctrl	I_ctrl	IDLE		
	11	I_ctrl	P_ctrl	PLOAD		
DATA4	00	data	data	DATA2		
	01	data	P_ctrl	PLOAD		
	01	data	I_ctrl	IDLE		
	10	P_ctrl	data	DATA1		
	11	I_ctrl	I_ctrl	IDLE		
	11	I_ctrl	P_ctrl	PLOAD		
DATA5	00	data	data	DATA3		
	01	data	P_ctrl	PLOAD		
	01	data	I_ctrl	IDLE		
	10	P_ctrl	data	DATA1		
	11	I_ctrl	I_ctrl	IDLE		
	11	I_ctrl	P_ctrl	PLOAD		

Figure 45

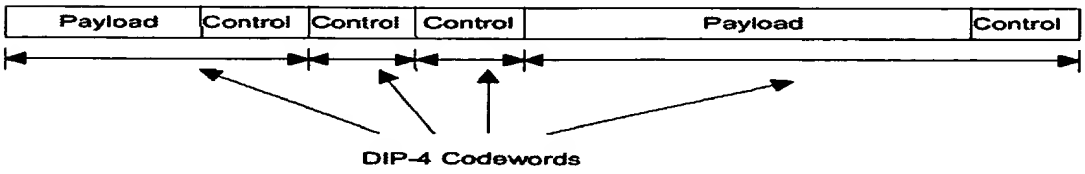


Figure 46

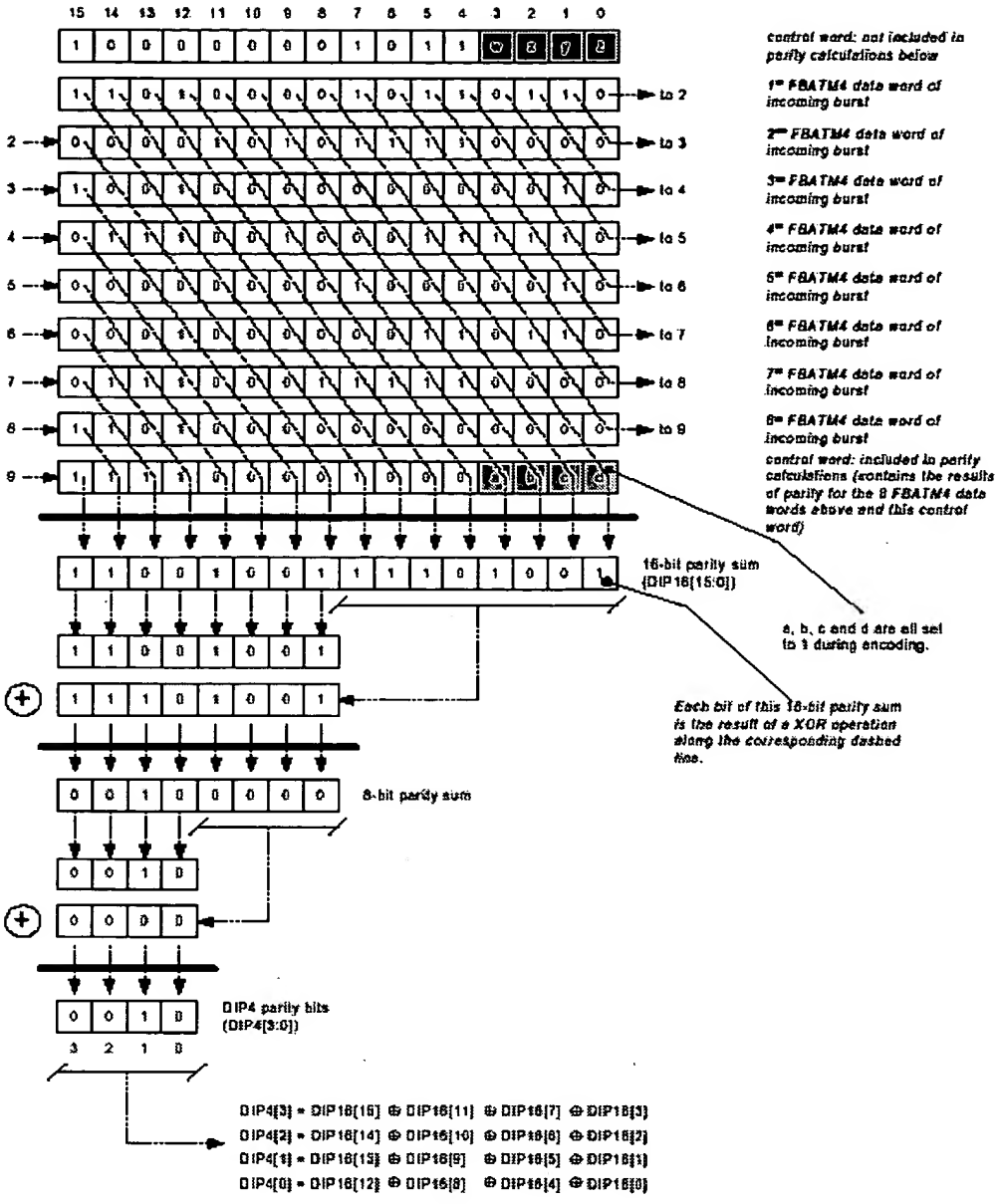

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
control word: not included in parity calculations below
1st FBATM4 data word of incoming burst
2nd FBATM4 data word of incoming burst
3rd FBATM4 data word of incoming burst
4th FBATM4 data word of incoming burst
5th FBATM4 data word of incoming burst
6th FBATM4 data word of incoming burst
7th FBATM4 data word of incoming burst
8th FBATM4 data word of incoming burst
control word: included in parity calculations (contains the results of parity for the 8 FBATM4 data words above and this control word)
16-bit parity sum (DIP16[15:0])
a, b, c and d are all set to 1 during encoding.
Each bit of this 16-bit parity sum is the result of a XOR operation along the corresponding dashed line.
8-bit parity sum
DIP4 parity bits (DIP4[3:0])
3 2 1 0
DIP4[3] = DIP16[15] ⊕ DIP16[11] ⊕ DIP16[7] ⊕ DIP16[3]
DIP4[2] = DIP16[14] ⊕ DIP16[10] ⊕ DIP16[6] ⊕ DIP16[2]
DIP4[1] = DIP16[13] ⊕ DIP16[9] ⊕ DIP16[5] ⊕ DIP16[1]
DIP4[0] = DIP16[12] ⊕ DIP16[8] ⊕ DIP16[4] ⊕ DIP16[0]

Figure 47

Example of CALENDAR[i] = 1, ..., CALENDAR_LEN

Two OC-48 channles (ports 1 and 2), eight OC-12 channles (ports 3 through 10):
CALENDAR_LEN = 16, CALENDAR[i] = 1, 2, 3, 4, 1, 2, 5, 6, 1, 2, 7, 8, 1, 2, 9, 10, ...
Other combinations feasible: CALENDAR[i] = 1, 3, 2, 4, 1, 5, 2, 6, 1, 7, 2, 8, 1, 9, 2, 10, ...

Time slot sequence →

1	2	3	4	1	2	5	6	1	2	7	8	1	2	9	10

CALENDER[i] is the round robin sequence of time slots among the lowest data rate channels. This example shows that each time slot coresponds to a OC-12 channel. One OC-48 port will be repeated 4 time slots within each CALENDAR sequence to have the proper data rate.

Figure 48

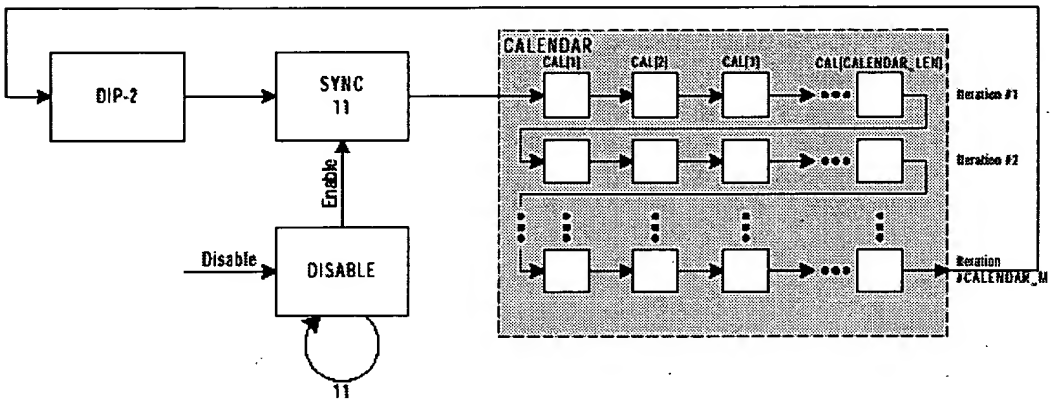


Figure 49

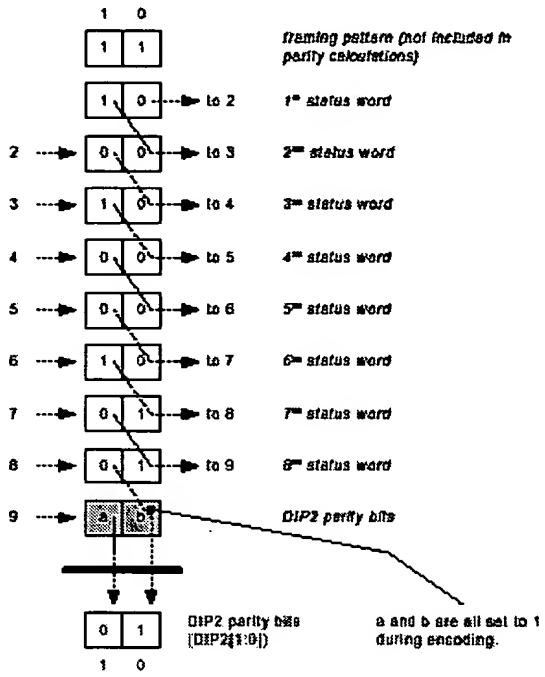


Figure 50

Name	*Type*	*Width*	*Depth*	*Total size*	*Access Time*
Input FIFO	SSRAM (dual-port)	34 bits	256	8704 bits	2.5ns read/write
Sync FIFO	SSRAM (dual-port)	64 bits	8	512 bits	5ns read/write
SPI Table	SSRAM (dual-port)	29 bits	64	1856 bits	5ns read/write
Data FIFO	SSRAM (dual-port)	64 bits	672	43008 bits	5ns read/write
Control FIFO	SSRAM (dual-port)	27 bits	84	2268 bits	5ns read/write
Free Buffer List	SSRAM (dual-port)	10 bits	84	840 bits	5ns read/write

Figure 51

Name	Description	Width	Dir	Clock
	Chip Pins			
RDCLK	Receive Data Clock (400Mhz)	1	In	Yes
RDAT	Receive Data	16	In	400 MHz
RCTL	Receive Control	1	In	400 MHz
RSCLK	Receive Status Clock	1	Out	Yes
RSTAT	Receive FIFO Status	2	Out	200 Mhz

RDCLK_L	Receive Data Clock (400Mhz)	1	In	Yes
RDAT_L	Receive Data	16	In	400 MHz
RCTL_L	Receive Control	1	In	400 MHz
RSCLK_L	Receive Status Clock	1	Out	Yes
RSTAT_L	Receive FIFO Status	2	Out	200 Mhz
Lookup Engine				
DATAOut	64 bit output data bus	64	Out	200 MHz
Data_valid	Valid signal	1	Out	200 MHz
SOP	Start of Packet	1	Out	200 MHz
EOP	End of Packet	1	Out	200 MHz
SOB	Start of the burst transfer	1	Out	200 MHz
EOB	End of the burst transfer	1	Out	200 MHz
PORT	PHY port addresses	6	Out	200 MHz
Byte_valid	Valid bytes within the last 64 bit word	3	Out	200 MHz
Discard	Pass the abort and parity error	1	Out	200 MHz
Segmentation				
Sg_spi_pid	Port number for the FIFO status bit	6	in	200 Mhz
Sg_spi_status	FIFO Status	1	In	200 Mhz
CPU Interface				

Figure 52

Register Name	Address	Description
CALENDAR_LEN		This is the length of the port sequence which is programmed upon start-up, depending on the number of active ports in the system; the maximum supported length is 64 ports.
CALENDAR_M		The number of times the calendar sequence to be repeated during the FIFO info transfers.

Figure 53

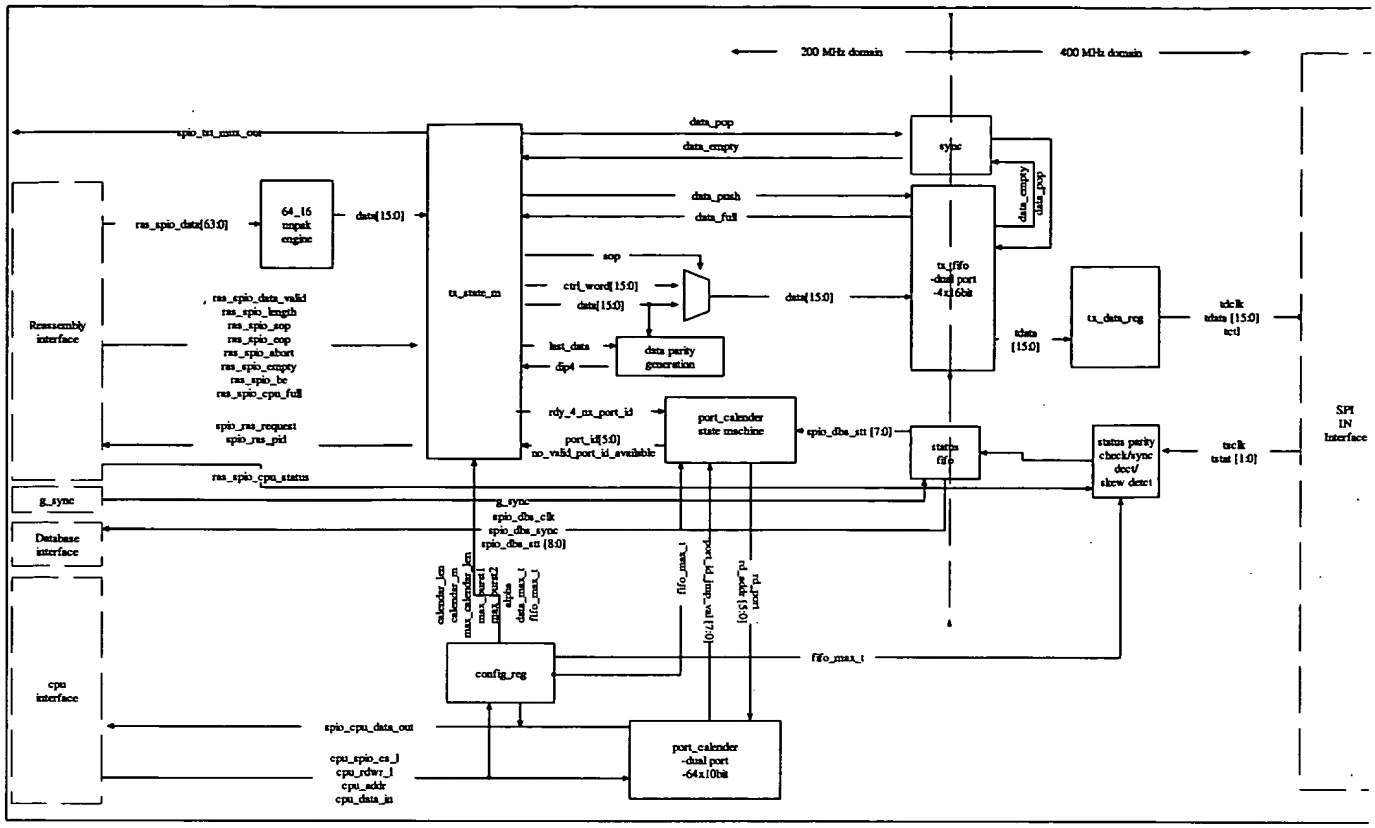


Figure 54

(Note: In cycle 1, XX and abcd depend on the contents of the interval after the last preceding control word.)

Cycle	TCTL / RCTL	TDAT[i] / RDAT[i]															
		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	0	X	X	0	0	0	0	0	0	0	0	0	a	b	c	d
2 - 11	1	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
12 - 21	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
20α -18 – 20α-9	1	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
20α-8 - 20α+1	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0

Figure 55

Current State	Data_ valid	Input	Next State	Word1	Word2	Pos_v	Neg_v	Comments
IDLE	0	X	IDLE	I_ctrl	I_ctrl	1	1	
	1	P_ctrl	PLOAD	I_ctrl	I_ctrl	1	1	
	0	T_ctrl	TRAIN_C	I_ctrl	I_ctrl	1	1	
TRAIN_C	0	T_ctrl	TRAIN_C	T_ctrl	T_ctrl	1	1	
	0	T_data	TRAIN_D	T_ctrl	T_ctrl	1	1	
TRAIN_D	0	T_data	TRAIN_D	T_data	T_data	0	0	
	1	P_ctrl	PLOAD	T_data	T_data	0	0	
	0	I_ctrl	IDLE	T_data	T_data	0	0	
	0	T_ctrl	TRAIN_C	T_data	T_data	0	0	
PLOAD	1	Data	DATA1	P_ctrl	Data	1	0	
	1	EOP	PLOAD	P_ctrl	Data	1	0	
	0	EOP	IDLE	P_ctrl	Data	1	0	
DATA1	1	Data	DATA3	Data	Data	0	0	
	1	EOP	DATA0	Data	P_ctrl	0	1	
	0	EOP	IDLE	Data	I_ctrl	0	1	
DATA2	1	Data	DATA0	Data	Data	0	0	
	1	EOP	DATA0	Data	P_ctrl	0	1	
	0	EOP	IDLE	Data	I_ctrl	0	1	
DATA3	1	Data	DATA1	Data	Data	0	0	
	1	EOP	DATA0	Data	P_ctrl	0	1	
	0	EOP	IDLE	Data	I_ctrl	0	1	
DATA0	1	Data	DATA2	Data	Data	0	0	
	1	EOP	DATA0	Data	P_ctrl	0	1	
	0	EOP	IDLE	Data	I_ctrl	0	1	

Figure 56

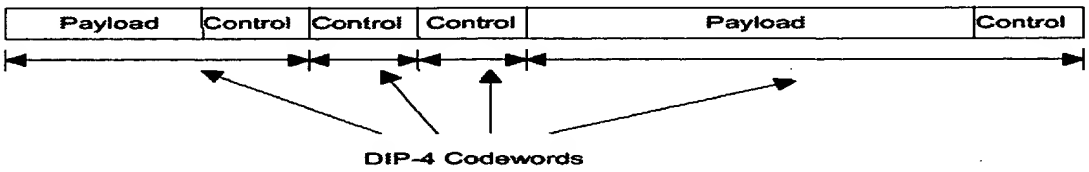


Figure 57

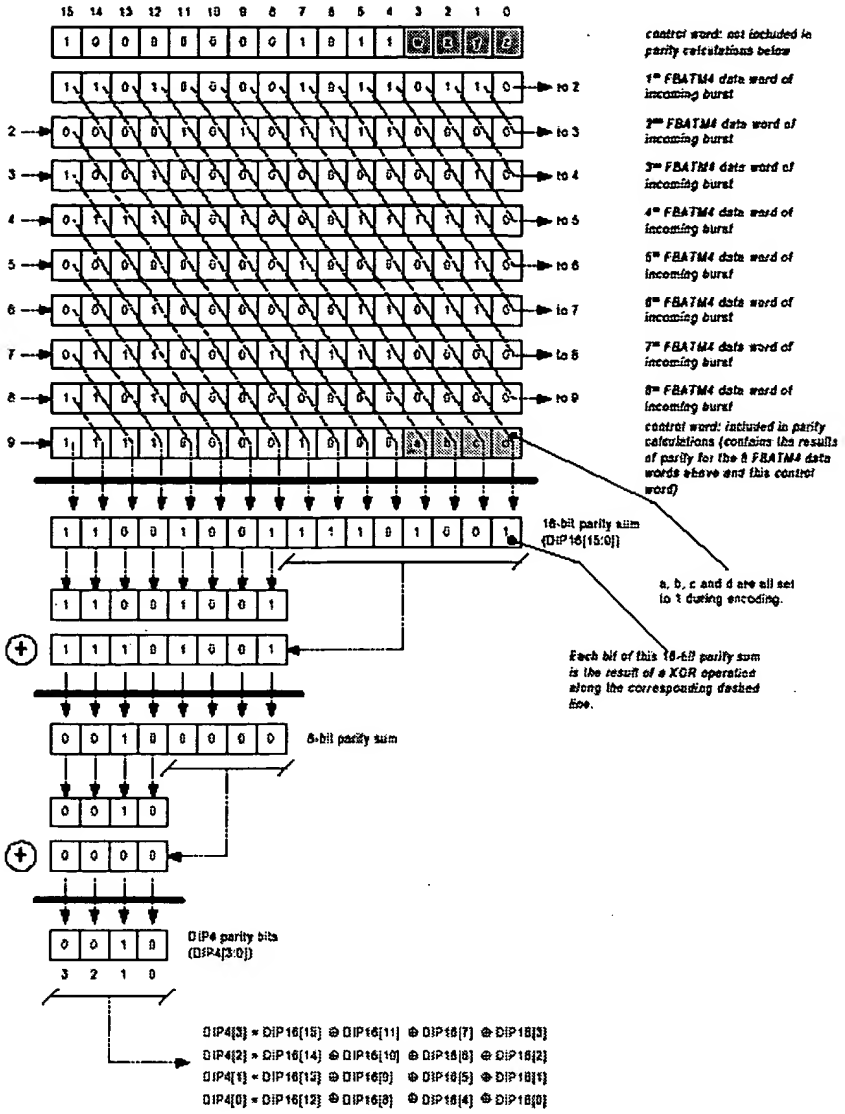


Figure 58

	Bit 15 Bit 8	Bit 7 Bit 0
Data Word1	Byte1	Byte2
Data Word2	Byte3	Byte 4
Data Word3	Byte5	Byte6
Data Word4	Byte7	Byte8

Figure 59

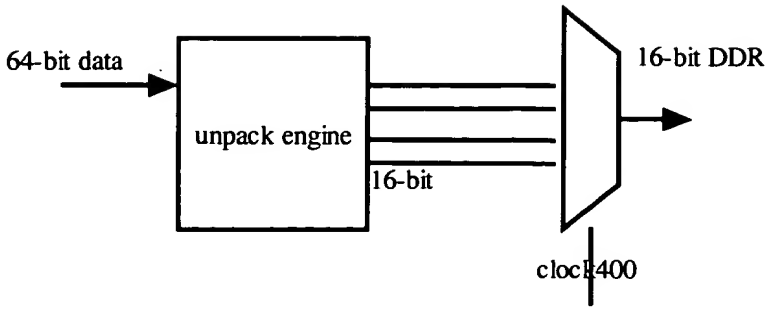


Figure 60

J: Jump indicates this is the last port on the calender
V: Valid indicates the port is valid for data transfer
Addr: equivalent of port entry number, range is 0 to 63

Event

1. System Reset

Addr	port ID	J	V
0	x	0	0
1	x	0	0
2	x	0	0
3	x	0	0
4	x	0	0
5	x	0	0
6	x	0	0
7	x	0	0

2. Add first port (5)

Addr	port ID	J	V
0	5	1	1
1	x	0	0
2	x	0	0
3	x	0	0
4	x	0	0
5	x	0	0
6	x	0	0
7	x	0	0

3. Add port 7

Addr	port ID	J	V
0	5	0	1
1	7	1	1
2	x	0	0
3	x	0	0
4	x	0	0
5	x	0	0
6	x	0	0
7	x	0	0

3. Add port 3

Addr	port ID	J	V
0	5	0	1
1	7	0	1
2	3	1	1
3	x	0	0
4	x	0	0
5	x	0	0
6	x	0	0
7	x	0	0

Figure 61

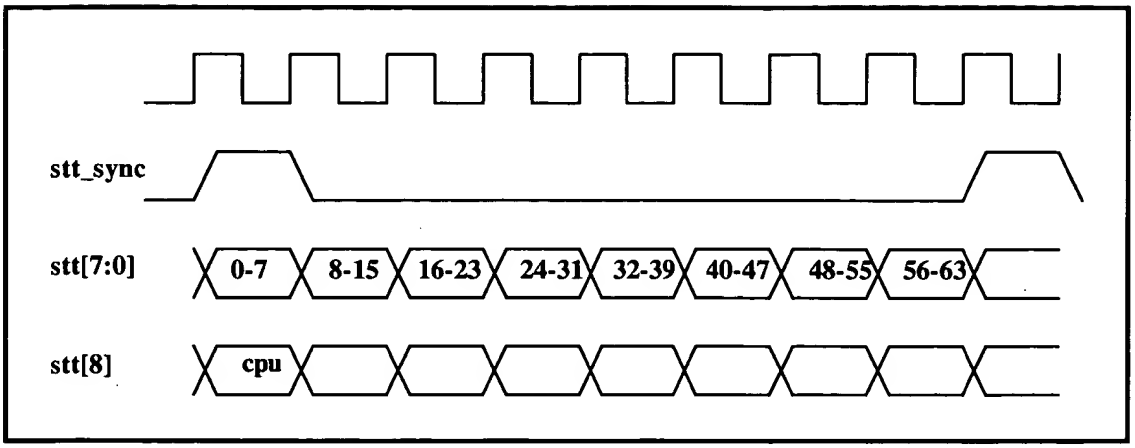


Figure 62

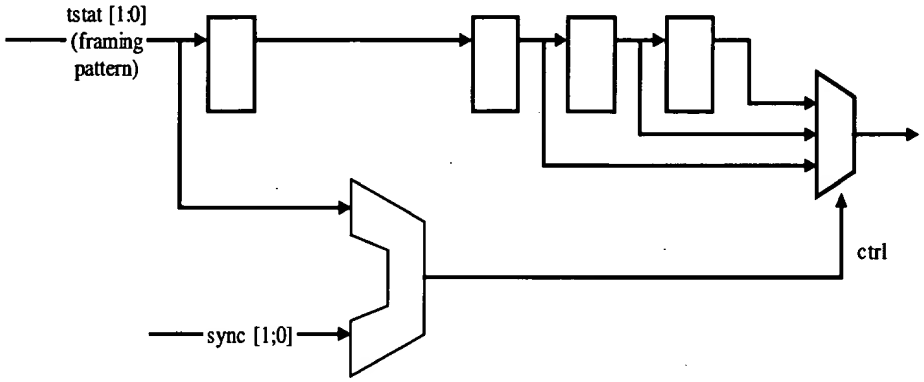


Figure 63

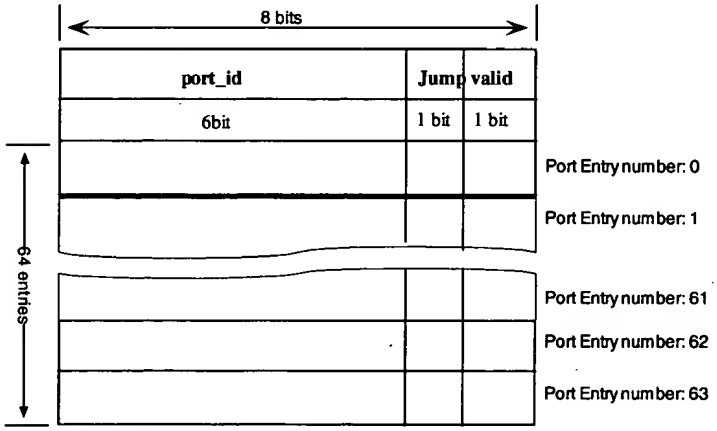


Figure 64

Name	Description	Width	Dir	Clock
Chip pins				
Tdclk	1	Out	Transmit Data Clock (400Mhz)	400 MHz
Tdat	16	Out	Transmit Data	400 MHz
Tctl	1	Out	Transmit Control	400 MHz
Tsclk	1	In	Transmit Status Clock	400 MHz
Tstat	2	In	Transmit FIFO Status	400 MHz
Re-assembly				
Ras_spio_data	64	In	64 bit input data bus	200 MHz
Ras_spio_data_valid	1	In	Valid signal	200 MHz
Ras_spio_sop	1	In	Start of Packet	200 MHz
Ras_spio_eop	1	In	End of Packet	200 MHz
Ras_spio_length	8	In	Data length	200 MHz
Ras_spio_abort	1	In	Re-assembly mark abort for the packet	200 MHz
Ras_spio_empty	1	In	Re-assembly mark port data is empty	200 MHz
Ras_spio_bv	8	In	Re-assembly mark byte is valid	200 MHz
spio_ras_ctrl_request	1	Out	Request data from port indicated by spio_ras_pid	200 MHz
spio_ras_ctrl_pid	6	Out	Request data from port when spio_ras_ctrl_request is asserted	200 MHz
ras_spio_cpu_full	1	In	CPU DATA Buffer is full	200 MHz
Database				
spio_dbs_clk	1	Out	Input clock from the re-assembly block (a different chip)	200 Mhz
spio_dbs_sync	1	Out	A sync indication, if set, start of message	200 Mhz
spio_dbs_stt	1	Out	Status of the output ports in the re-assembly block.	200 Mhz
CPU Interface				
Spio_cpu_data_out	32	Out	Data from SPI to CPU	200 MHz
Cpu_spio_cs_l	1	In	Chip select	200

				MHz
Cpu_rdwr_l	1	In	Read/Write select.	200 MHz
Cpu_addr	20	In	CPU address	200 MHz
Cpu_data_in	32	In	CPU data	200 MHz
Global				
G_SYNC	1	In	This is the global sync from the GlobalSync block.	200MHz
spio_tst_mux_out	32	OUT	SPIO Test Mux pins.	200MHz
Spio_clk	1	IN	400 MHz system clock to SIPO block.	400MHz
spio_rst_l	1	IN	A combination of POR and SPIO Soft Reset.	200MHz
spio_in_en	1	IN	SPIO Input Enable signal.	200MHz
spio_out_en	1	IN	SPIO Output Enable Signal.	200MHz

Figure 65

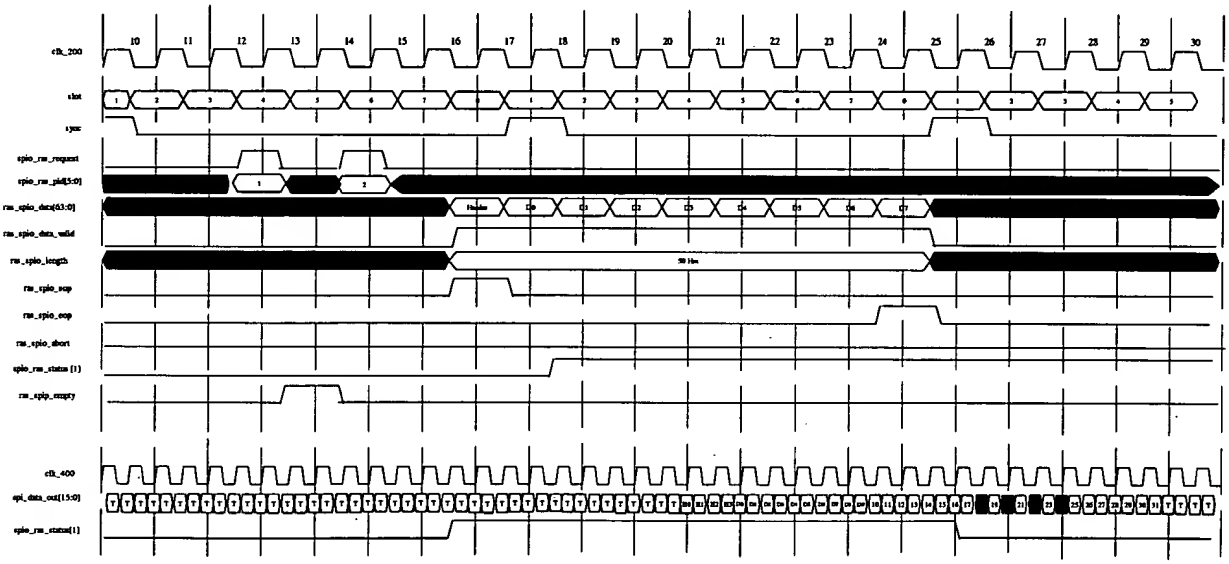


Figure 66

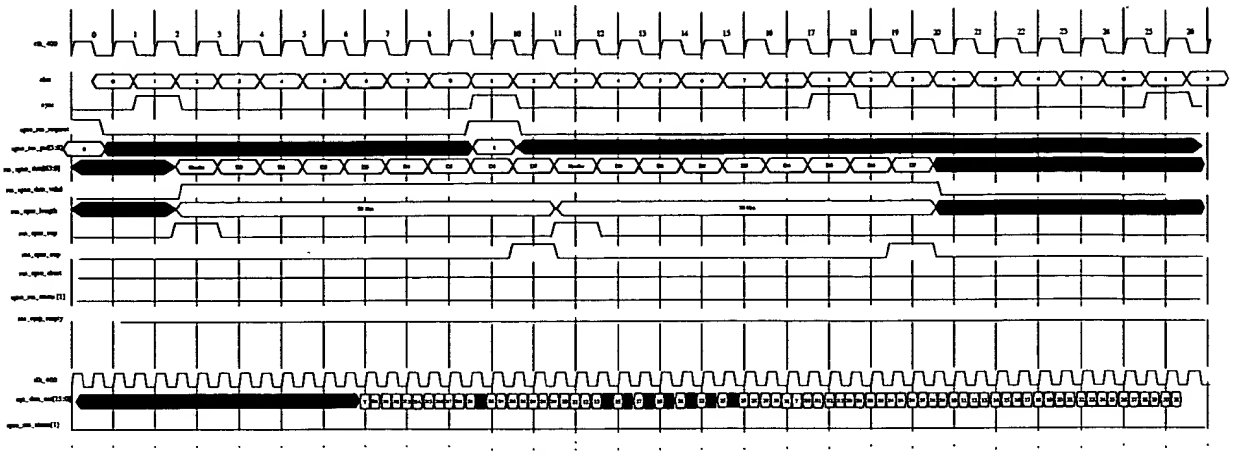


Figure 67

Address	Name	Type	Description
0	calender_len	RW	[15:0] – calender_len Corresponds to the number of ports with the lowest data rate that can be accommodated in the total data rate of the given application. CALENDAR_LEN must be at least as large as the number of active ports in the system.
0	calender_m	RW	[31:16] – Calendar sequence repeat M times The calendar sequence (of length CALENDAR_LEN) is repeated CALENDAR_M times before the DIP-2 parity and "1 1" framing words are inserted.
1	Max_clendar_len	RW	[15:0] – Maximum supported value of calendar_len. Indicates upper bound limit of the calendar_len. Users must ensure that the value of CALENDAR_LEN on the sending side of a FIFO status channel must not exceed MAX_CALENDAR_LEN on the receiving side.

1	Max_burst1	RW	[31:16] – FIFO can accept 16 byte blocks when FIFO status channel indicates starving. The value of the Max_burst1 must be greater than 5. Setting the value of Max_burst1 be greater than 5 is equivalent to setting burst size to be greater than 80 bytes (e.g. 96 bytes) of data per transfer. By setting minimum transfer size to be greater than 80 bytes, one entire Maximus cell (80 bytes) can be transfer in a given transfer without breaking one cell into two transfers. Thus avoid partial cell transfer.
2	Max_burst2	RW	[15:0] – FIFO can accept 16 byte blocks when FIFO status channel indicates starving. The value of the Max_burst2 must be greater than 5. Setting the value of Max_burst2 be greater than 5 is equivalent to setting burst size to be greater than 80 bytes (e.g. 96 bytes) of data per transfer. By setting minimum transfer size to be greater than 80 bytes, one entire Maximus cell (80 bytes) can be transfer in a given transfer without breaking one cell into two transfers. Thus avoid partial cell transfer.
2	Data_train_rep	RW	[31:16] – Number of repetitions of the data training. NOTE: the value of Data_train_rep should be minimum of 5. Since this parameter determines the frequency of data training pattern, by setting the value less than 5 may reduce the data bandwidth near or less than 10Gb/sec.
3	Data_max_t	RW	[15:0] – Interval between training sequences on data path interface.

3	Fifo_max_t	RW	[31:16] – Interval between training sequences on FIFO status path interface. NOTE: the value of FIFO_max_t should not exceed 64. Since this parameter determines the frequency of training pattern and parity, by setting the value less than 32 allows one parity check in approximately every 8 cycles (200 MHz).

Figure 68

Address	Name	Type	Description
0	COM	R/W	[31:28] – Opcode [27:0] – Address, depending on the command. No default value.
1-8	R0-R7	R/W	General-purpose register. No default value
7	R8	R/W	Port flush indicator for ports 31 to 0. Bit location [0] corresponds to port 0, and bit [1] corresponds to port 1,etc. When the bit is set to 1'b1, the corresponding port should ignore the SPI status from the SPI interface and send "not full" status to the Database block. Also, when the bit is set to 1'b1, SPI should mark the data as bad while sending data out for the given port. The purpose of the flush bit is to flush data left in the pipe for a given port so the data left from a dead port does not remain inside the memory.
8	R9	R/W	Port flush indicator for ports 64 to 32. Bit location [0] corresponds to port 32, and bit [1] corresponds to port 33,etc. See R6 for description.
9– 31	Reserved		
32	CONTROL	R/W	[0] – MODE. If set, Outgoing SPI is operating in an Ingress chip. If reset the Outgoing SPI is operating in an Egress chip Default value 0 (egress mode). [1] – OUT_EN Global output enable for all ports, if reset, no output stage will be performed for all ports Default value 0 (output disabled). [31:2] – Reserved
33	R33 (spio_err)	R	If there is a parity error, the error is reported into the control status register (spio_err [0]) by setting 1'b1 into the bit. This spio_err register is self-cleared upon CPU read.

34	SPIO_TSTMUX_ SEL	RW	[3:0] – Testmux_Selection: 0000: No output (default) 0001: Group 1 = {TBD} 0010: Group 2 = {TBD} 0011: Group 3 = {TBD} 0100: Group 4 = {TBD} 0101: Group 5 = {TBD} 0110: Group 6 = {TBD} 0111: Group 7 = {TBD} 1000: Group 8 = {TBD} 1001: Group 9 = {TBD} 1010: Group 10 = {TBD} 1011: Group 11 = {TBD} 1100: Group 12 = {TBD} 1101: Group 13 = {TBD} 1110: Group 14 = {TBD} 1111: Group 15 = {TBD}
35 - 63	Reserved		

Figure 69

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00001	R[17:0]				Addr[9:0]		
R0	calender_m				calender_len			

Figure 70

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00010	R[17:0]				Addr[9:0]		
R0	calender_m				calender_len			

Figure 71

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00011	R[17:0]				Addr[9:0]		
R1	Max_burst1				Max_clendar_len			

Figure 72

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00100	R[17:0]				Addr[9:0]		
R1	Max_burst1				Max_clendar_len			

Figure 73

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00101	R[17:0]				Addr[9:0]		
R2	Data_train_rep				Max_burst2			

Figure 74

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01000	R[17:0]				Addr[9:0]		
R3	Fifo_max_t				Data_max_t			

Figure 75

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01001	R[19:0]					Addr[7:0]	
R5	Reserved		Reserved		Port Entry number		Port ID + J+V	

P – Port ID
J – Jump bit
V – Port valid

Figure 76

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01010	R[19:0]					Addr[7:0]	
R5	Reserved		Reserved		Port Entry number		Port ID + J+V	

P – Port ID
J – Jump bit
V – Port valid

Figure 77

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01011	R[19:0]					Addr[7:0]	
R6	Port flush for port [31:0]							
R7	Port flush for port [63:32]							

Figure 78

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01100	R[19:0]					Addr[7:0]	
R6	Port flush for port [31:0]							
R7	Port flush for port [63:32]							

Figure 79

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01101	R[19:0]					Addr[7:0]	
R33	Spio_err							

Figure 80

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01110	R[19:0]					Addr[7:0]	
R34	SPIO_TSTMUX_SEL							

Figure 81

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01111	R[19:0]					Addr[7:0]	
R34	SPIO_TSTMUX_SEL							

Figure 82

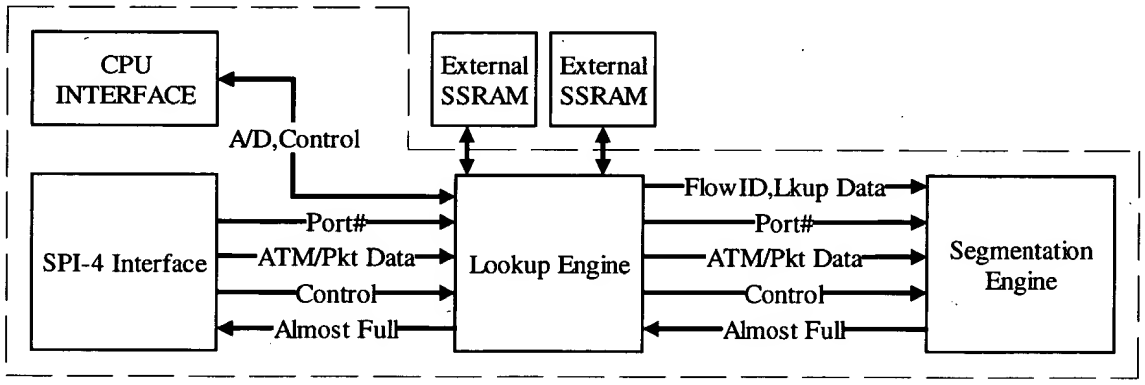


Figure 83

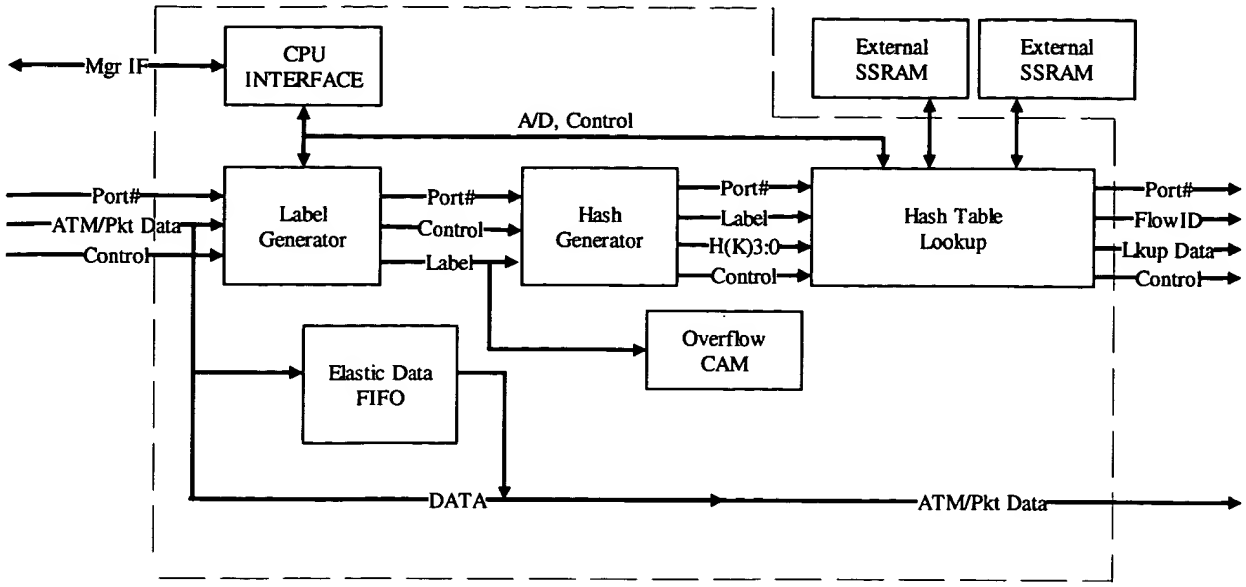


Figure 84

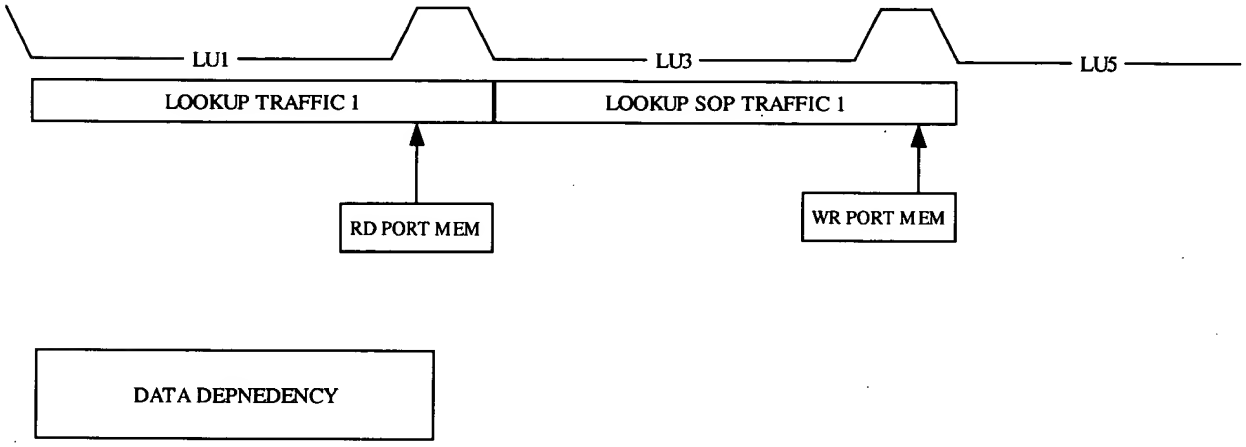


Figure 85

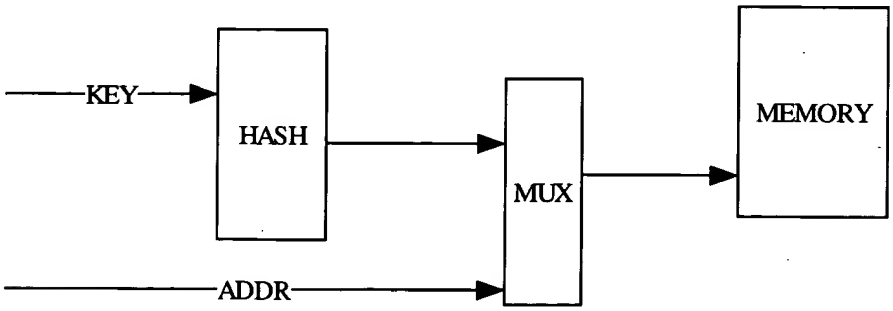


Figure 86

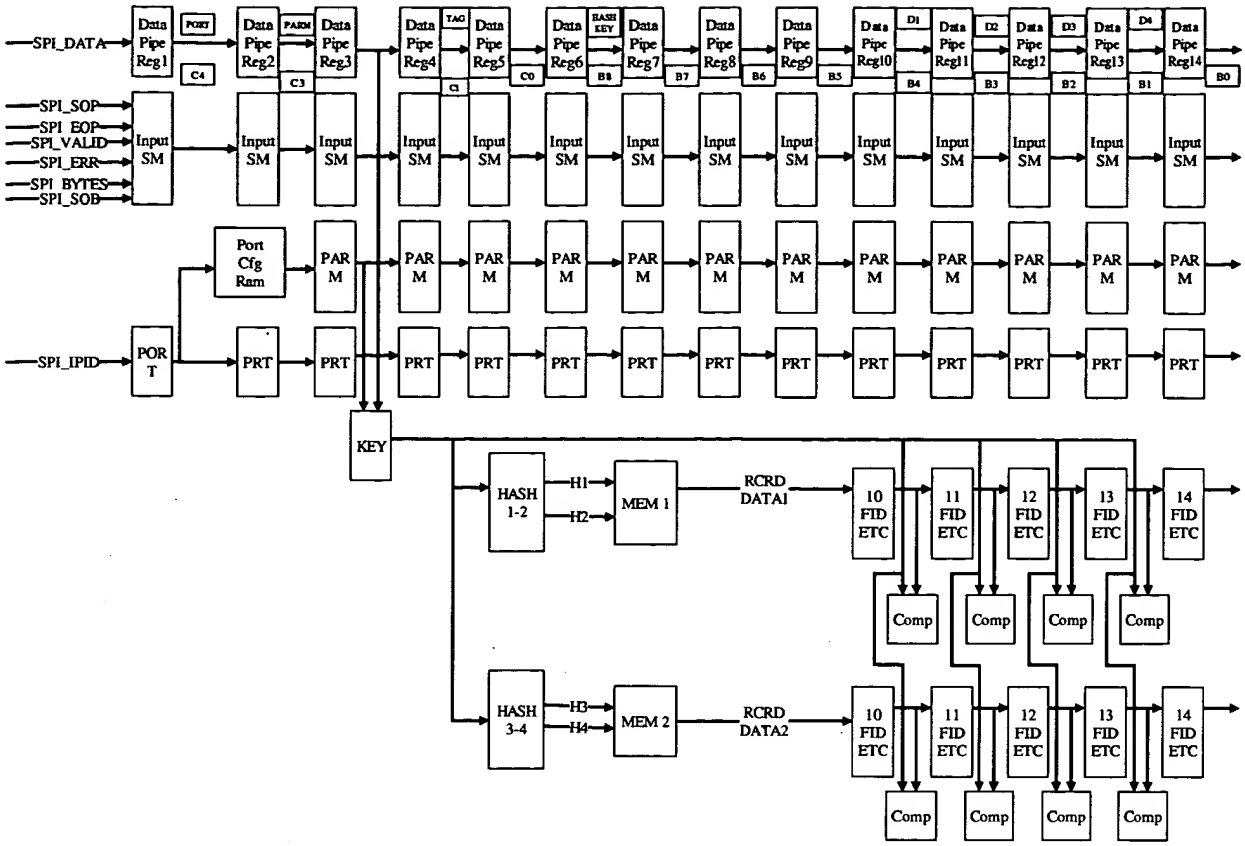


Figure 87

NAME	WIDTH	RANGE	TYPE	WR	DESCRIPTION
PORT TYPE	3	2:0	ALL	S	The lookup type of incoming traffic of the FID
Complete INT & SW HD STRIP	5	7:3	DIR/DEF	S	The number of bytes to end of header to strip that includes the switch header and the internal header
INT HDR EXTRACT	5	12:8	DIR/DEF	S	The number of bytes to the start of Internal header starting from the end of the complete INT & SW header
FID	20	32:13	ALL	H/S	FID used until EOP or always for default type
FID TYPE	4	3 6:33	ALL	H/S	The FID type used until EOP or always for default type
EFCI	1	37	Default	H/S	EFCI BIT from SOP burst
CLP	1	38	Default	H/S	CLP BIT from SOP burst
OAM	1	39	Default	H/S	OAM BIT from SOP burst
CLASS	3	42:40	ALL	H/S	Class of traffic
RSVD	1	43	ALL		RSVD

Figure 88

NAME	WIDTH	BYTE	RANGE	WR	DESCRIPTION
CRC	1	0	0	S	If set, then generate the CRC
LENGTH	5	0	5:1	S	The number of bytes add
RSVD	2	0	7:6	S	N/A
ENCAP HEADER	128	17:1	135:8	S	HEADER to add

Figure 89

PORT TYPE	OPERATION	DESCRIPTION
000	None	Direct Flow ID (packets only)
001	No Hash	Default Flow ID
010	ALL	Special C mode
011	ALL	MPLS (PPP, Frame Relay over SONET 0
100	ALL	ATM (12-bitVPI, VCI)
101	ALL	ATM (8-bitVPI, VCI)
110	ALL	ATM (12-bit VPI only, mask out VCI)
111	ALL	ATM (8-bit VPI only, mask out VCI)

Figure 90

NAME	NO BITS	RANGE	WR	DESCRIPTION
LABEL or TAG	34	33:0	S	This is the label that will be compared to the input to the hash table
Valid	1	34	S	If set, then the entry is valid
FID	20	54: 35	S	
Traffic Type	4	58:55	S	If set, then the head BID is the EOP one for packets.
CLASS	3	61:59	S	Not used
RSVD	10	71:62		Not Used

Figure 91

NAME	Width in Bits
DATA	64
FID	20
TYPE	4
CLASS	1
EFCI	1
CLP	1
OAM	1
CRC	1
SOP	1
EOP	1
SOB	1
EOB	1
PORT	6
VALID BYTE	3
VALID	1

Figure 92

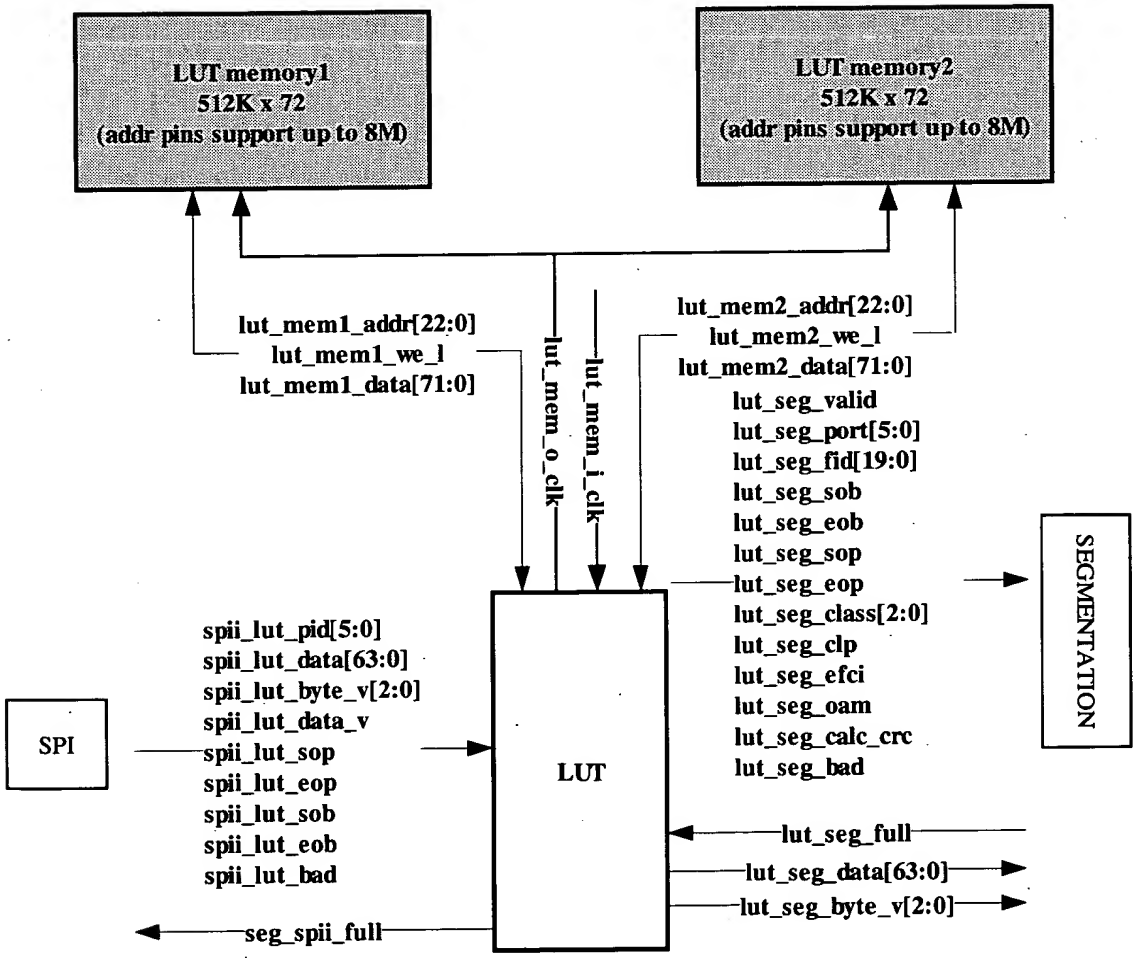


Figure 93

SIGNAL NAME	WIDTH	DIR	DESCRIPTION
SPI Interface			
SPI_DATA	64	IN	Input data bus
SPI_VALID	1	IN	Input data valid signal
SPI_SOP	1	IN	Start of Packet
SPI_EOP	1	IN	End of Packet
SPI_SOB	1	IN	Start of data burst
SPI_EOB	1	IN	End of data burst
SPI_IPID	6	IN	Input port ID (IPID)
SPI_BYTES	3	IN	# Of valid bytes in the last data word
SPI_ERR	1	IN	Marks bad packets for the segmentation engine
SPI_ALMOST_FULL	1	OUT	Nearly Full
Segmentation Interface			
LKUP_DATA	64	OUT	Output data bus
LKUP_VALID	1	OUT	Output data valid signal
LKUP_SOP	1	OUT	Start of Packet
LKUP_EOP	1	OUT	End of Packet
LKUP_SOB	1	OUT	Start of data burst
LKUP_PID	6	OUT	Input port ID (PID)
LKUP_BYTES	3	OUT	# Of valid bytes in the last data word
LKUP_ERR	1	OUT	Marks bad packets for the segmentation engine
LKUP_TYPE	4	OUT	Type of traffic
LKUP_FID	20	OUT	Flow ID
LKUP_OAM	1	OUT	OAM
LKUP_CLP	1	OUT	CLP
LKUP_EFCI	1	OUT	EFCI
LKUP_CRC	1	OUT	Calculate CRC
SEG_ALMOST_FULL	1	In	Nearly Full

Figure 94

SIGNAL	DESCRIPTION
POR_L	Power on reset.
RST_LU_L	Reset the Look Up engine. It is a level signal.
INPUT_ENABLE	Input block enable. It is zero on power up.
OUTPUT_ENABLE	Output block enable. It is zero on power up.

Figure 95

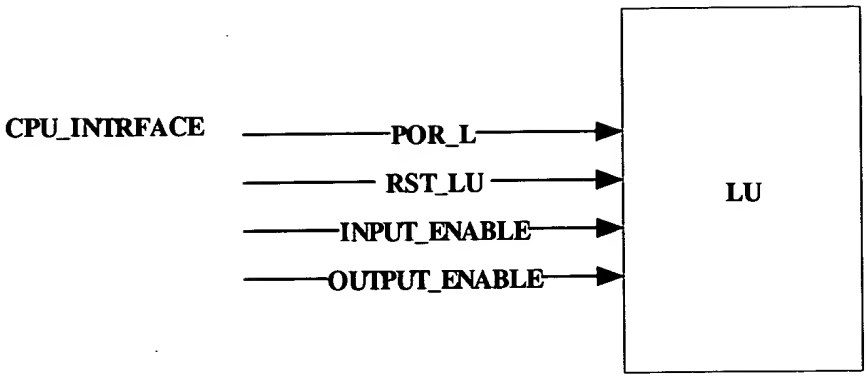


Figure 96

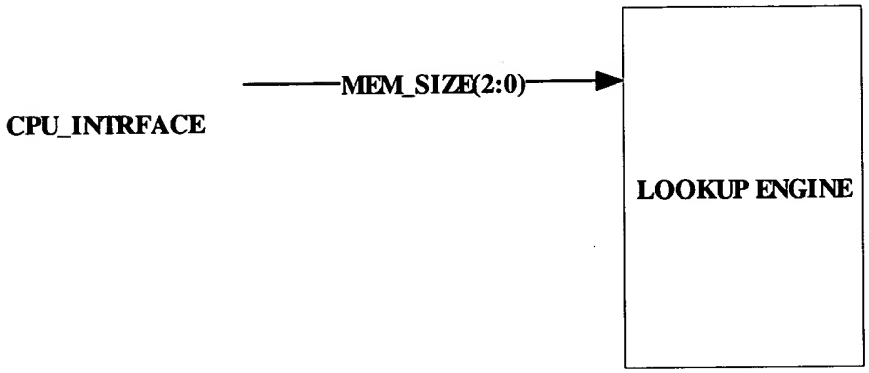


Figure 97

MEM_SIZE INPUT VALUES	MEMORY SIZE
001	1 M
001	2 M
010	3 M
011	4 M
100	5 M
101	5 M
110	7 M
111	8 M

Figure 98

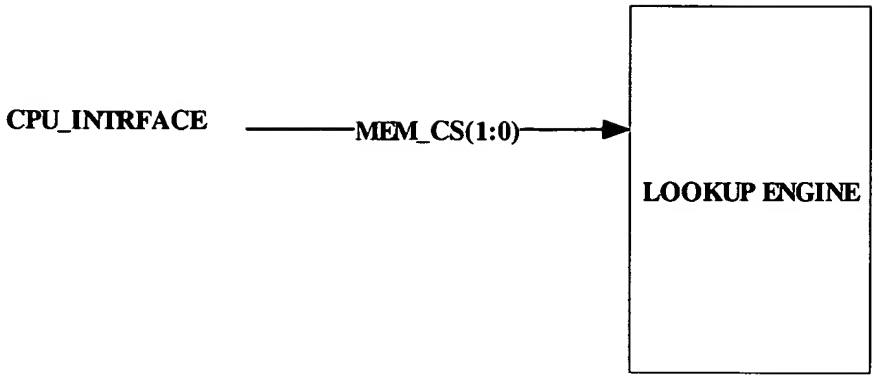


Figure 99

MEM_CS	ADRESS_BITS
00	20:19
01	21:20
10	22:21
11	Not Used

Figure 100

Start address	Last address	Total length
40h	3Fh	64d

Figure 101

Address	Name	Type	Description
0	Command + Address	R/W	[31:28] – Opcode, [27:0] - Address
1	R0	R/W	[31:0] 32 bits of data
2	R1	R/W	[63:32] 32 bits of data
31-3	Reserved		
32	CPU FID register	R/W	[19:0] 20 bit wide CPU FID that is used when there is no match
33	Memory Channel Enable	R/W	[31:2] Reserved If set enable Memory Channel 1 to access memory 1 and CAM 1 during lookup, Default value "1". [1] If set enable Memory Channel 0 to access memory 0 and CAM 0 during lookup, Default value "1". [0]
63-33	Reserved		Not Used

Figure 102

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00000	Don't Care						
R0	Don't Care							
R1	Don't Care							

Figure 103

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00001	Don't Care		ADDR [22:0]				
R0	R0 [31:0] Data							
R1	R1 [31:0] Data							
R2	R2 [7:0] Data							

Figure 104

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00010	Don't Care		ADDR [22:0]				
R0	R0 [31:0] Data							
R1	R1 [31:0] Data							
R2	R2 [7:0] Data							

Figure 105

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00001	Don't Care		ADDR [22:0]				
R0	R0 [31:0] Data							
R1	R1 [31:0] Data							
R2	R2 [7:0] Data							

Figure 106

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00010	Don't Care		ADDR [22:0]				
R0	R0 [31:0] Data							
R1	R1 [31:0] Data							
R2	R2 [7:0] Data							

Figure 107

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00011	Don't Care		ADDR [5:0]				
R0	R0 [31:0] Data							
R1	R1 [11:0] Data							

Figure 108

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00100	Don't Care		ADDR [5:0]				
R0	R0 [31:0] Data							
R1	R1 [11:0] Data							

Figure 109

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01001	Don't Care						
R0	LUT_KEY [31:0], Register Bits (31:0)							
R1	LUT_KEY [33:32], Register Bits (1:0) Valid, Register Bit (2) FID [19:0], Register Bits (22:3) LUT_TYPE [3:0], Register Bits (26:23) LUT_CRC_KEY_SEL [1:0], Register Bits (28:27) [CLASS [2:0], Register Bits (31: 29)							

Figure 110

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01010	Don't Care						
R0	LUT_KEY [31:0], Register Bits (31:0)							
R1	LUT_KEY [33:32], Register Bits (1:0) LUT_CRC_KEY_SEL [1:0], Register Bits (3:2) [DONTCARE], Register Bits (31:4)							

Figure 111

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01011	Don't Care						
R0	LUT_KEY [31:0], Register Bits (31:0)							
R1	LUT_KEY [33:32], Register Bits (1:0) LUT_CRC_KEY_SEL [1:0], Register Bits (3:2) [DONTCARE], Register Bits (31: 4)							

Figure 112

R0	R0 [22:0] Data R0 [31:23] reserved

Figure 113

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01100	Don't Care		ADDR [4:0]				
R0	R0 [31:0] Data							
R1	R1 [31:0] Data							
R2	R2 [7:0] Data							

Figure 114

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01101	Don't Care		ADDR [4:0]				
R0	R0 [31:0] Data							
R1	R1 [31:0] Data							
R2	R2 [7:0] Data							

Figure 115

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01110	Don't Care		ADDR [4:0]				
R0	R0 [31:0] Data							
R1	R1 [31:0] Data							
R2	R2 [7:0] Data							

Figure 116

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01111	Don't Care		ADDR [4:0]				
R0	R0 [31:0] Data							
R1	R1 [31:0] Data							
R2	R2 [7:0] Data							

Figure 117

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0

Figure 118

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	10001	Don't Care						

Figure 119

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	10010	Don't Care						

Figure 120

EFCI	1	7
CLP	1	6
OAM	1	5

Figure 121

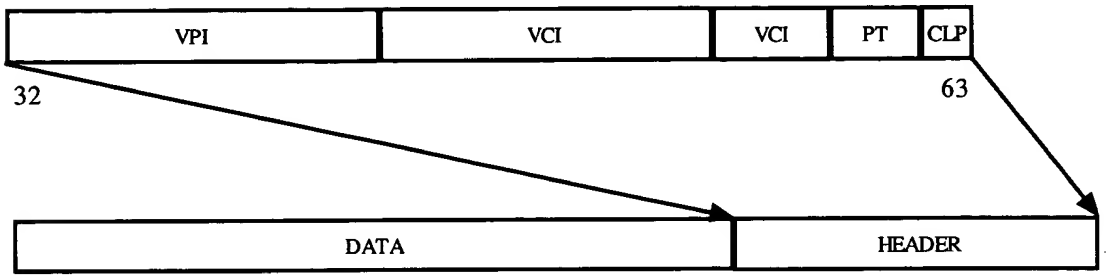


Figure 122

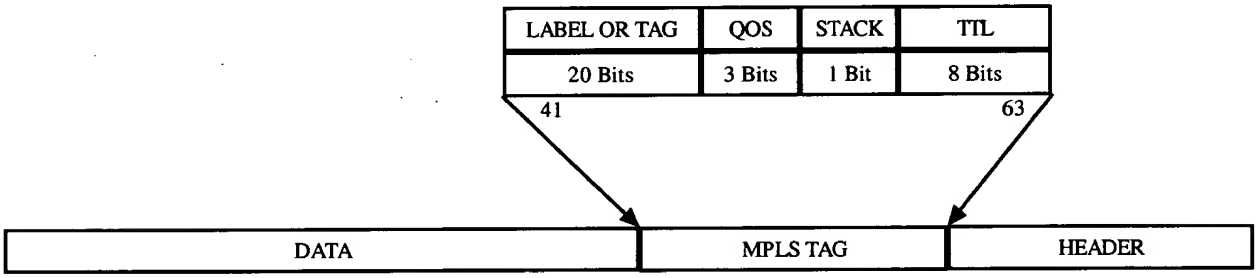


Figure 123

PORT TYPE	OPERATION	DESCRIPTION
000	None	Direct Flow ID (packets only)
001	No Hash	Default Flow ID
010	ALL	Special C mode
011	ALL	MPLS (PPP, Frame Relay over SONET 0
100	ALL	ATM (12-bitVPI, VCI)
101	ALL	ATM (8-bitVPI, VCI)
110	ALL	ATM (12-bit VPI only, mask out VCI)
111	ALL	ATM (8-bit VPI only, mask out VCI)

Figure 124

NAME	BITS	RANGE
FID	20	12:31
TYPE	4	8:11
EFCI	1	7
CLP	1	6
OAM	1	5
CLASS	3	2:4
RSVD	2	0:1

Figure 125

SOP	EOP	CLASS	OAM	CLP	EFCI	TYPE	FID
1 bit	1 bit	3 bits	1 bit	1 bit	1 bit	4 bits	20 bits
0	1	0:3				0:3	0:19
0	1	2:4	5	6	7	8:11	12:31

Figure 126

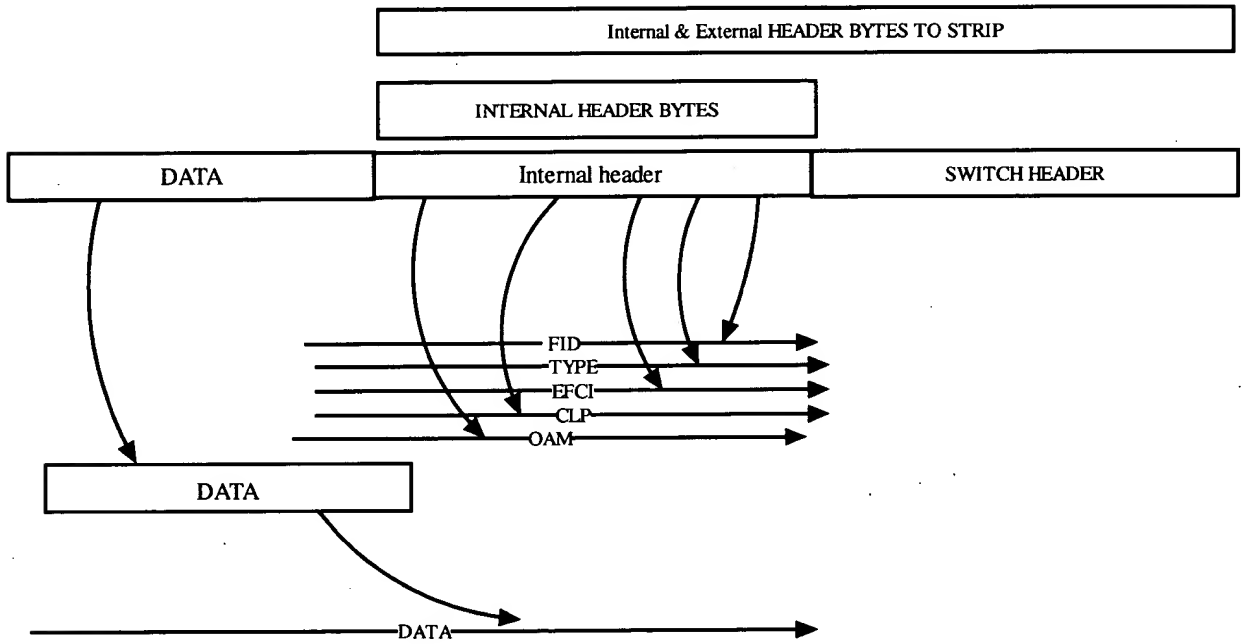


Figure 127

SPCL LK	HDR ADDR	RSVD	OAM	CLP	EFCI	TYPE	CLASS	CID
1 bit	3 bits	1 bit	1 bit	1 bit	1 bit	4 bits	3 bits	17 bits
0	1:3	4	5	6	7	8:11	12:14	15:31

Figure 128

CRC	LENGTH	RSVD	ENCAPSULATION HEADER
1	5 Bits	2	16 BYTES

Figure 129

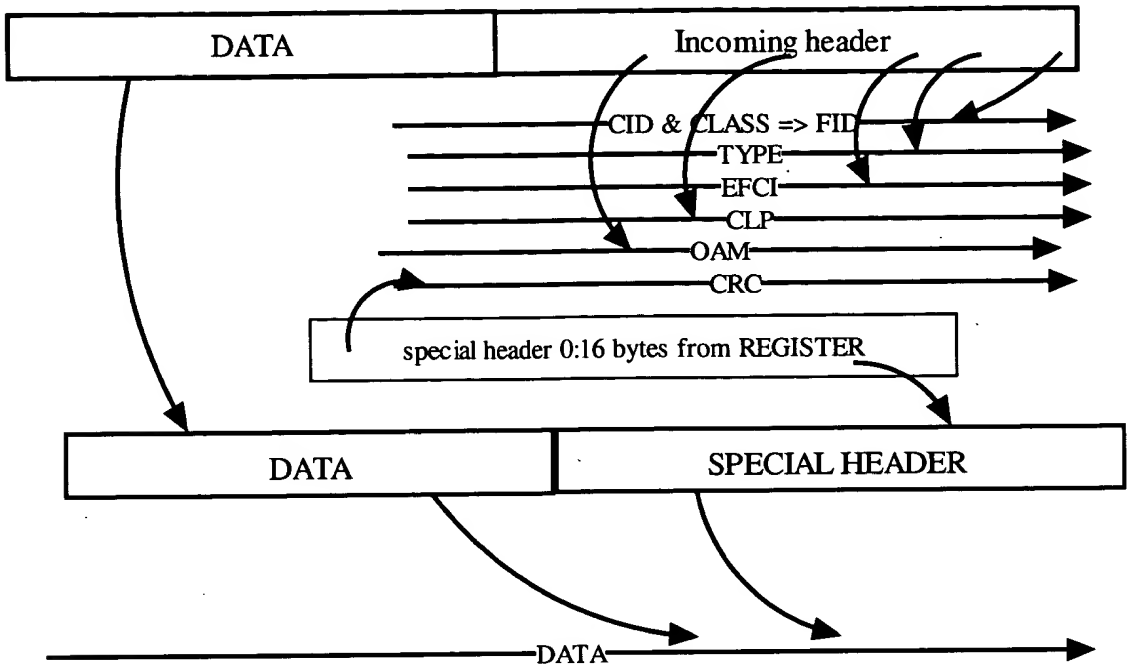


Figure 130

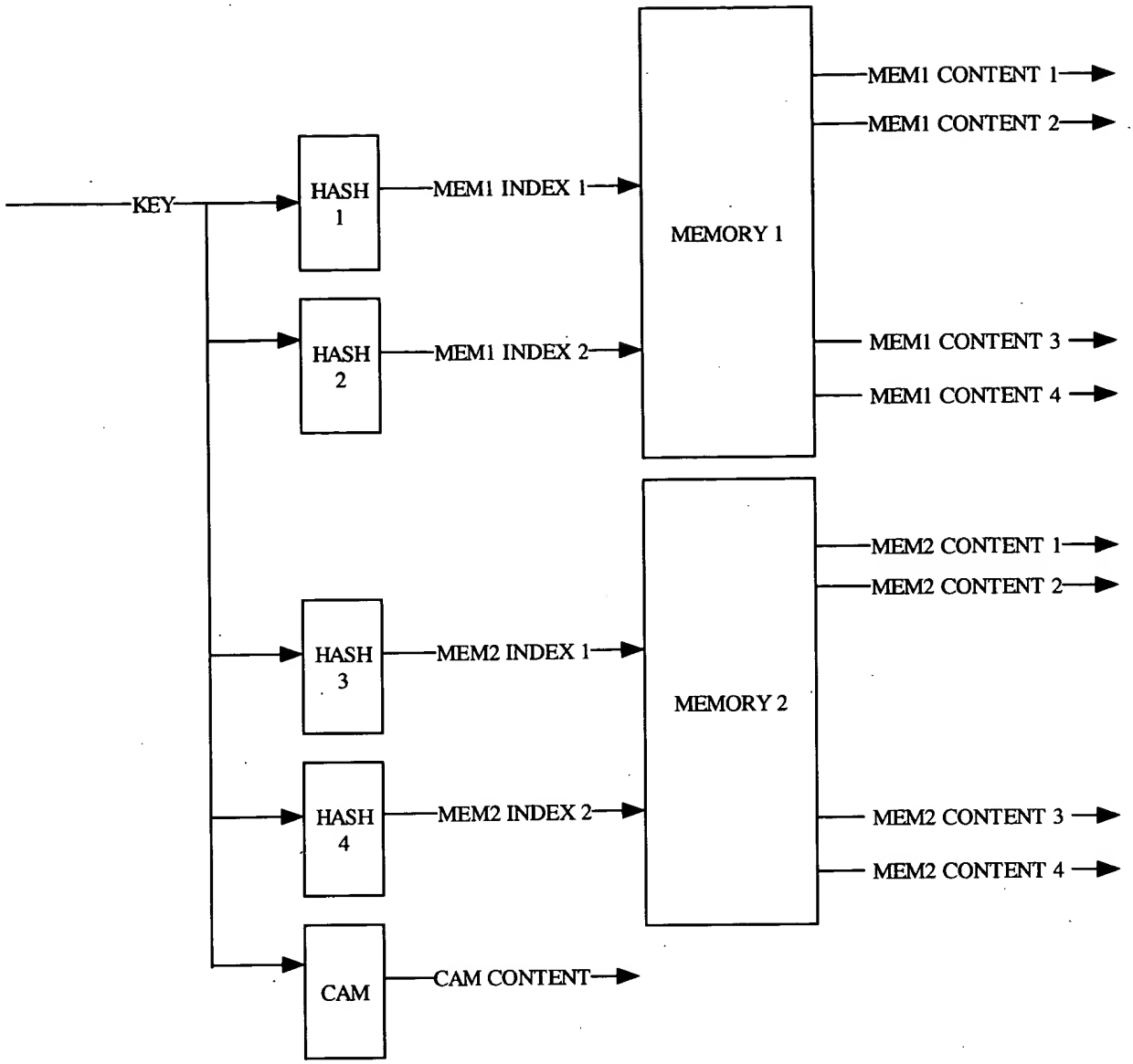

KEY
HASH 1
HASH 2
HASH 3
HASH 4
CAM
MEM1 INDEX 1
MEM1 INDEX 2
MEM2 INDEX 1
MEM2 INDEX 2
CAM CONTENT
MEMORY 1
MEMORY 2
MEM1 CONTENT 1
MEM1 CONTENT 2
MEM1 CONTENT 3
MEM1 CONTENT 4
MEM2 CONTENT 1
MEM2 CONTENT 2
MEM2 CONTENT 3
MEM2 CONTENT 4

Figure 131

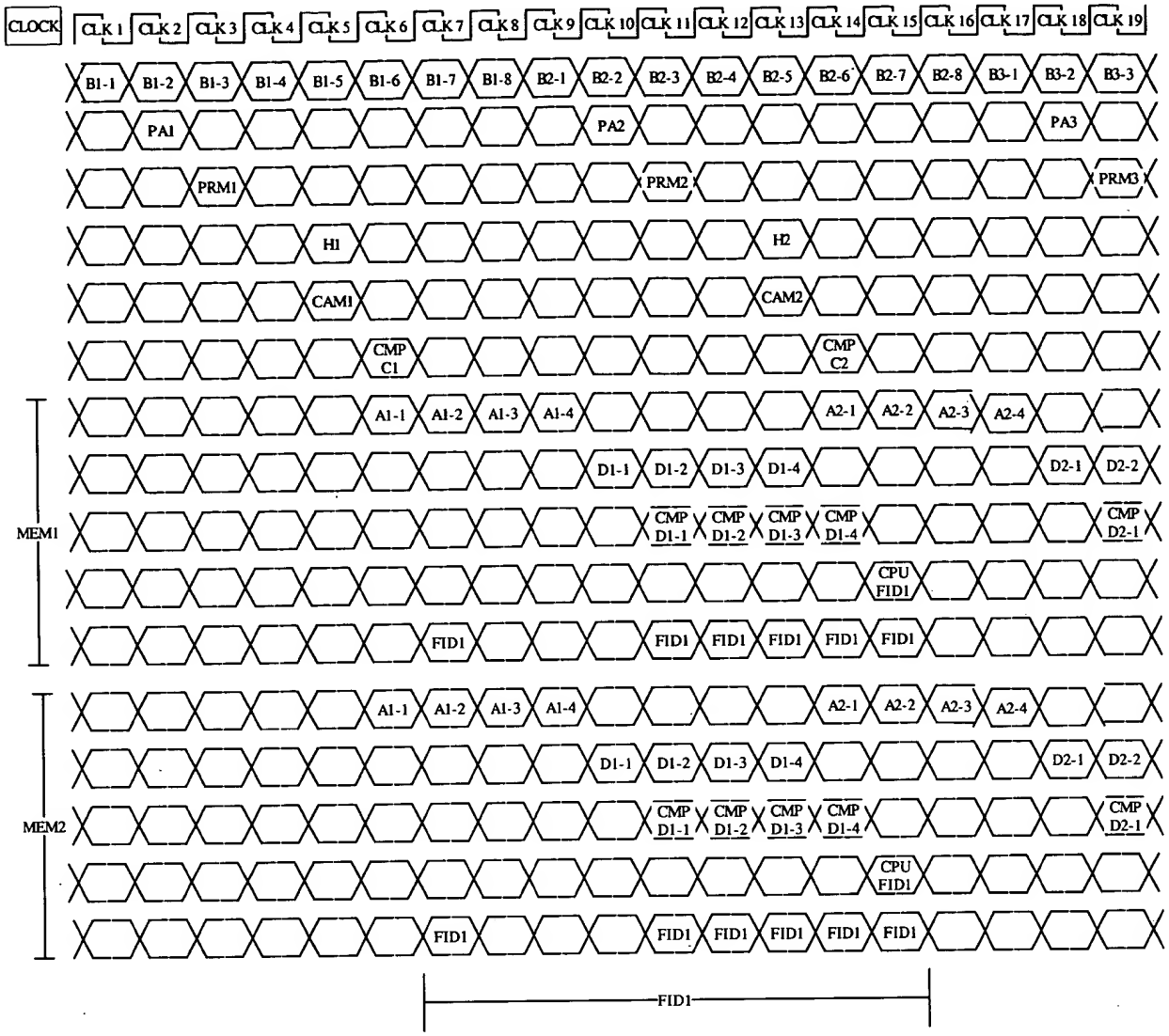

CLOCK
CLK 1
CLK 2
CLK 3
CLK 4
CLK 5
CLK 6
CLK 7
CLK 8
CLK 9
CLK 10
CLK 11
CLK 12
CLK 13
CLK 14
CLK 15
CLK 16
CLK 17
CLK 18
CLK 19
B1-1
B1-2
B1-3
B1-4
B1-5
B1-6
B1-7
B1-8
B2-1
B2-2
B2-3
B2-4
B2-5
B2-6
B2-7
B2-8
B3-1
B3-2
B3-3
PA1
PA2
PA3
PRM1
PRM2
PRM3
H1
H2
CAM1
CAM2
CMP C1
CMP C2
MEM1
A1-1
A1-2
A1-3
A1-4
A2-1
A2-2
A2-3
A2-4
D1-1
D1-2
D1-3
D1-4
D2-1
D2-2
CMP D1-1
CMP D1-2
CMP D1-3
CMP D1-4
CMP D2-1
CPU FID1
FID1
MEM2
FID1

Figure 132

SEG TYPE	TYPE	Application	SPII Input	SPII OUTPUT	LUT OUTPUT	Operation	SEG OUTPUT	CRC 32
1	0	Ingress TM (ATM => ATM)	ATM Cells 52B	52B 12B pad	52B ATM 12B pad	Pass cells	52B + 12B pad (ATM)	Off
2	1	Ingress TM (ATM => MPLS PKT)	ATM Cells 52B	52B 12B pad	52B ATM 12B pad	Remove 4B hdr Add 4B pad	48B + 16B pad (AAL5)	Off
3	2	Ingress TM (MPLS PKT => ATM)	PKT Bursts N x 16B	64B	64B PKT	Segment AAL5 Add 16B pad	48B + 16B pad (AAL5)	On
4	3	Ingress TM (PKT => PKT)	PKT Bursts N x 16B	64B	64B PKT	Segment 64B	64B Cells (AAL5 like)	On
1	4	ATM Encapsulation	ATM Cells 52B	52B 12B pad	52B ATM 12B pad	Pass cells	52B + 12B pad (ATM)	Off
2	5	Reassembly	ATM Cells 52B	52B 12B pad	52B ATM 12B pad	Remove 4B hdr Add 4B pad	48B + 16B pad (ATM)	Off
4	6	Ingress PKT Bypass	PKT Bursts N x 16B	64B	64B PKT	Segment 64B	64B Cells (AAL5 like)	On
5	7	Status Cell					64B cells	Off
1	8	Egress TM (ATM => ATM)	Switch HDR Switch Cells N x 16B	swx hdr ATM Up to 80B	52B ATM 12B pad	Pass cells	52B + 12B pad (ATM)	Off
1	9	Egress TM (ATM => MPLS PKT)	Switch HDR Switch Cells N x 16B	swx hdr AAL5 Up to 80B	48B AAL5 16B pad	Pass cells	48B + 16B pad (AAL5)	Off
1	10	Egress TM (MPLS PKT => ATM)	Switch HDR Switch Cells N x 16B	swx hdr AAL5 Up to 80B	48B AAL5 16B pad	Pass cells	48B + 16B pad (AAL5)	Off
1	11	Egress TM (PKT => PKT)	Switch HDR Switch Cells N x 16B	swx hdr AAL5 Up to 80B	64B AAL5	Pass cells	64B (AAL5)	Off
1	12	ATM De-Encapsulation	Switch HDR Switch Cells N x 16B	swx hdr ATM Up to 80B	52B ATM 4B pad	Add 8B pad	52B + 12B pad (ATM)	Off
3	13	Segmentation	Packets Bursts N x 16B	swx hdr PKT Up to 80B	64B PKT Up to 88B for L2	Calculate CRC for new L2 hdr Segment AAL5 Add 16B pad	48B + 12B pad (AAL5)	On
1	14	Egress PKT Bypass	Packets Bursts N x 16B	swx hdr AAL5 Up to 80B	64B AAL5	Pass cells	64B (AAL5)	Off
	15	Reserved						

Figure 133

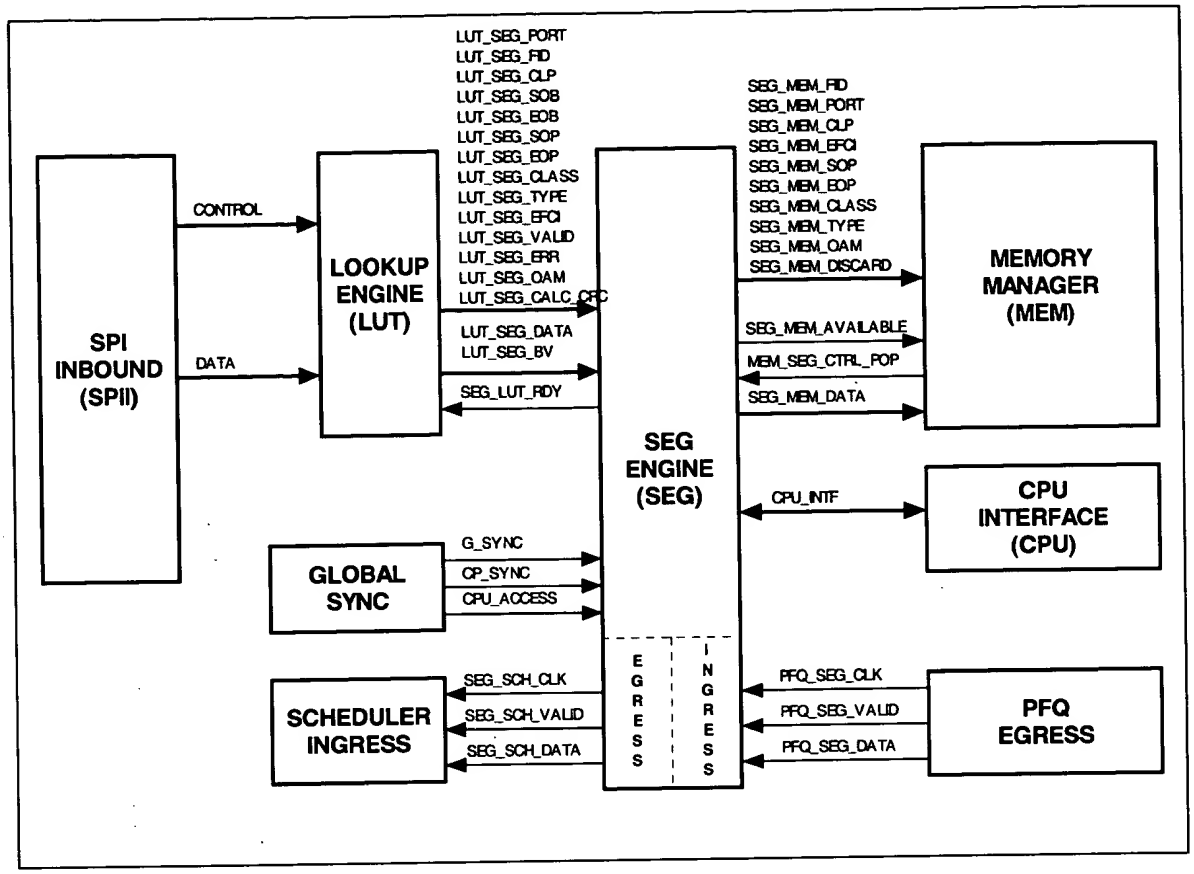


Figure 134

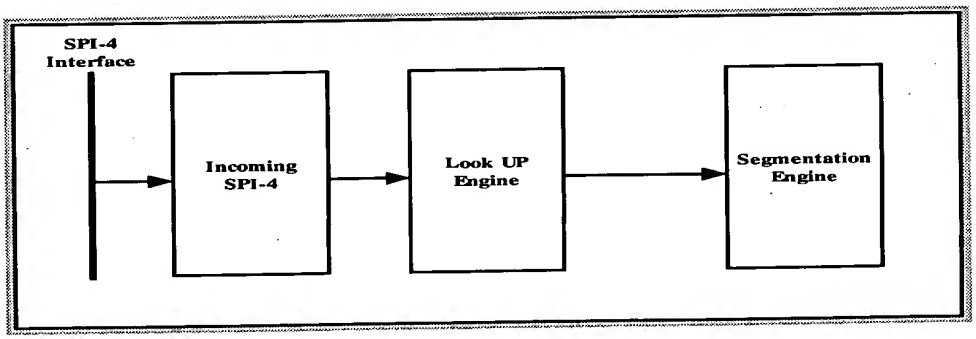


Figure 135

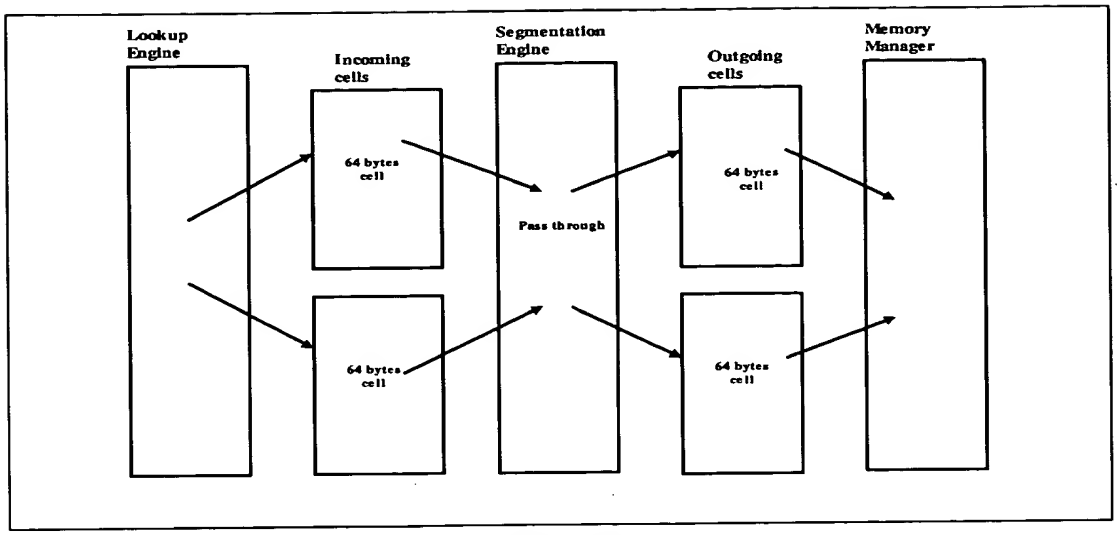


Figure 136

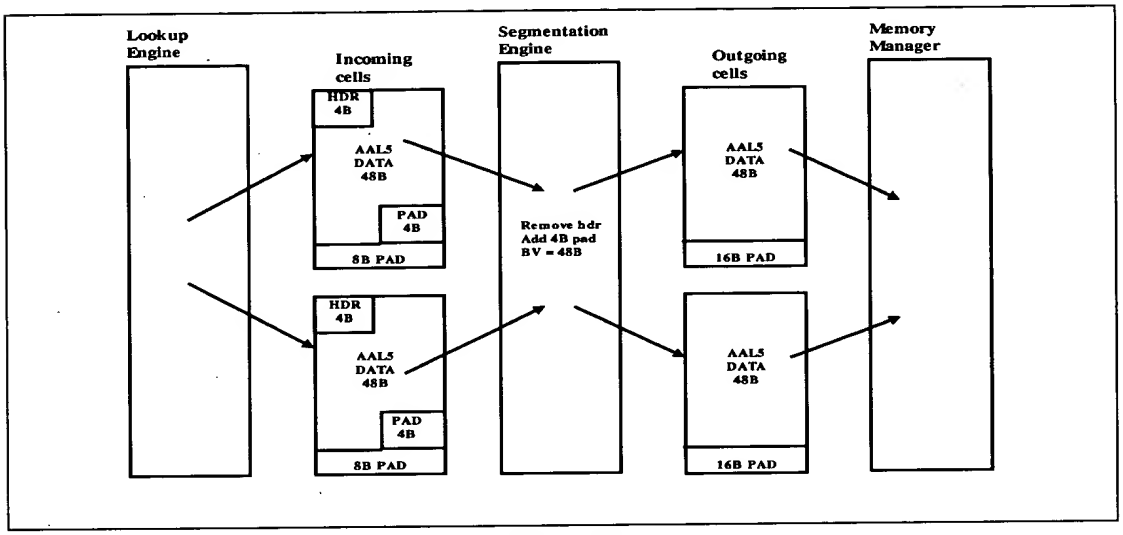


Figure 137

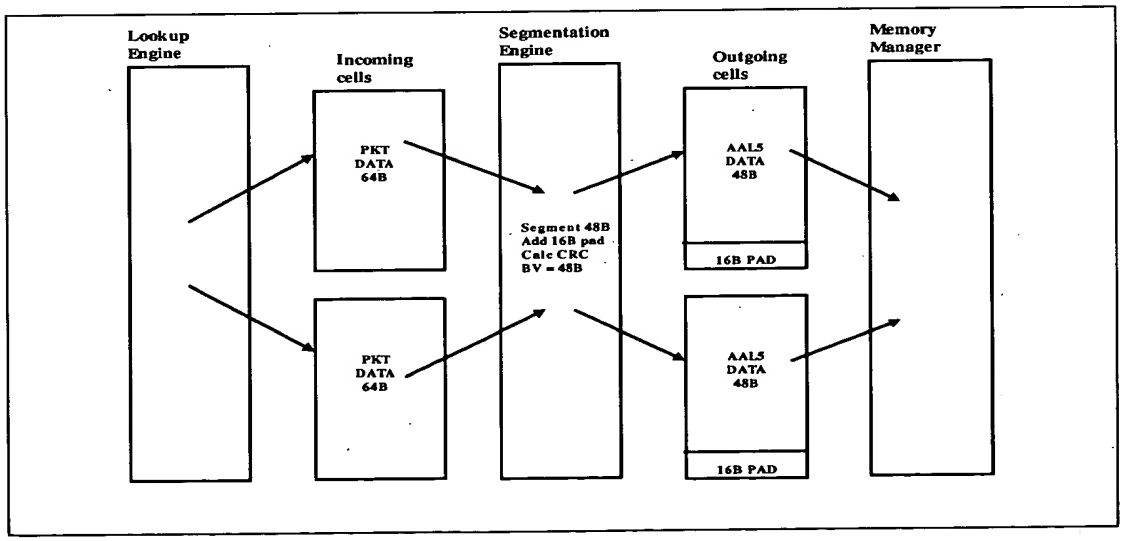


Figure 138

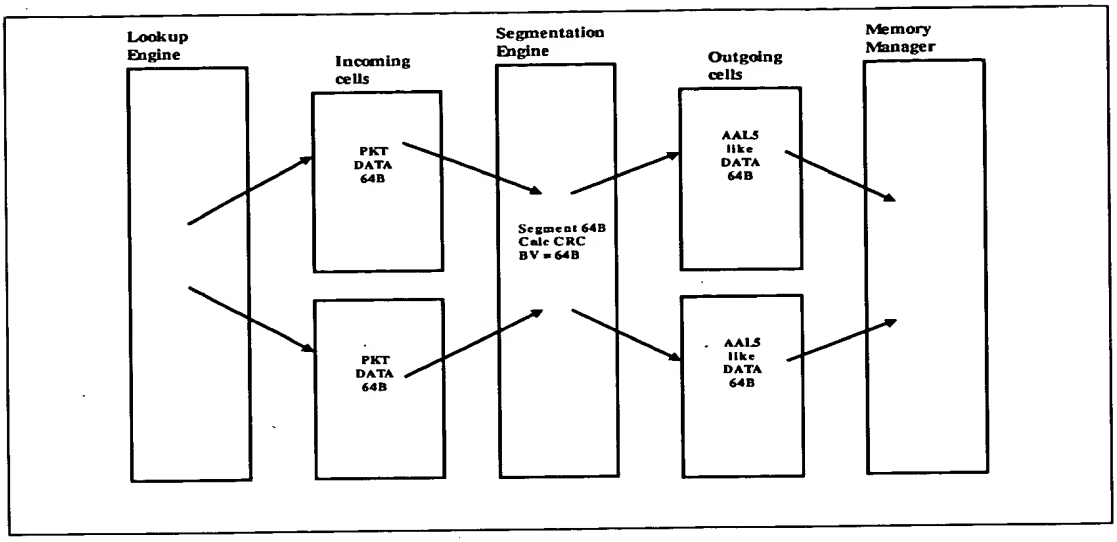


Figure 139

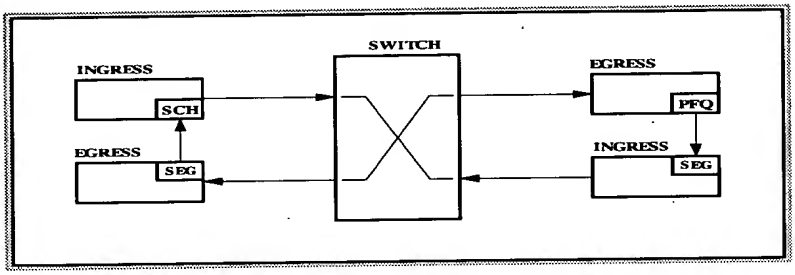


Figure 140

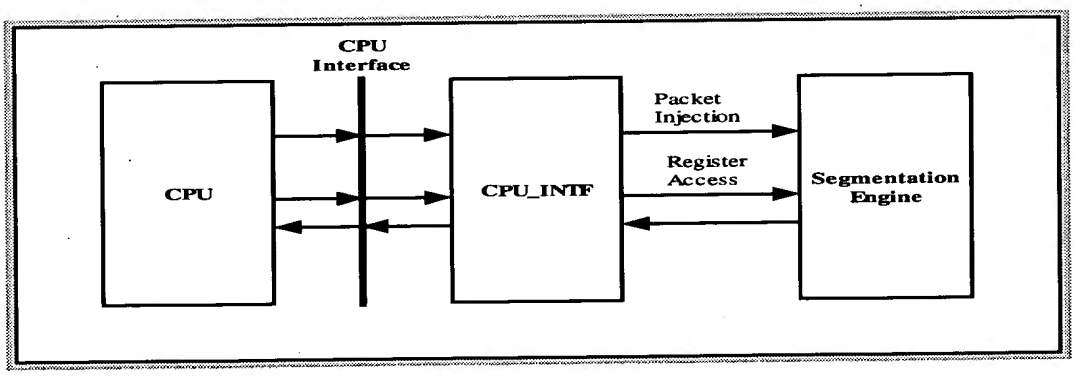


Figure 141

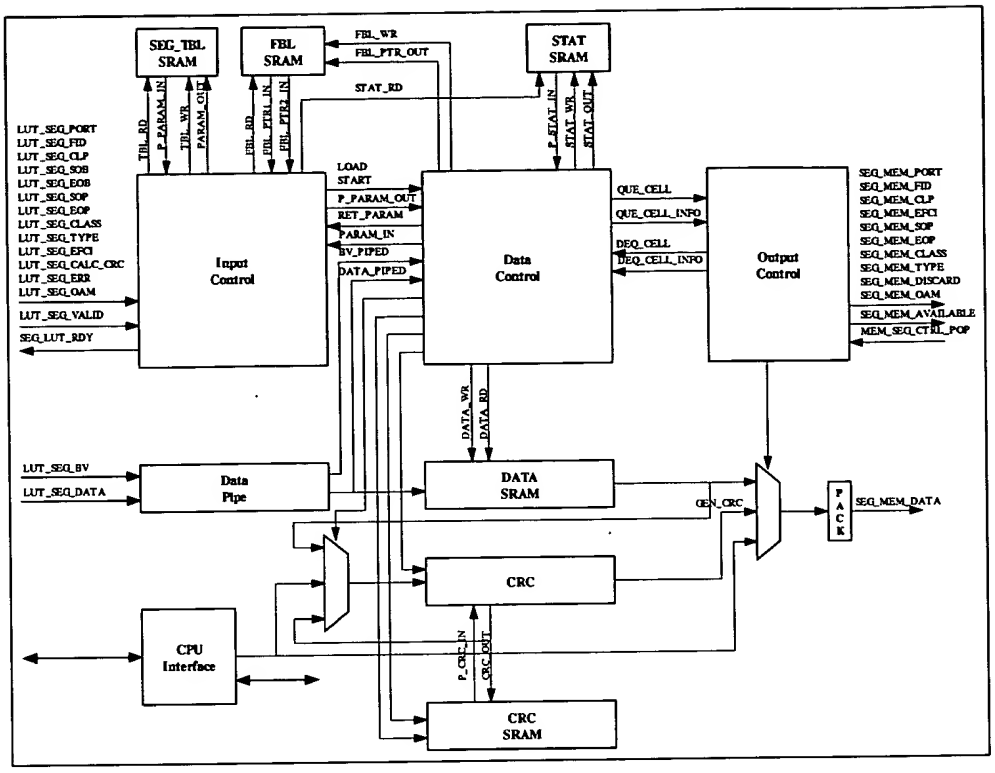


Figure 142

Errors	Description
SOP,SOP,EOP same port	This error happens when 2 SOPs are received with no EOP between them for the same port.
EOP,EOP,SOP same port	This error happens when 2 EOPs are received with no SOP between them for the same port.
Exceeds MTU	This error happens when the packet length is larger than the programmable maximum transfer unit.
PKT_LEN Error	This happens when the calculated packet length does not match the one in the trailer.
SEG full	The segmentation is full, i.e. no room in queue or data SRAM The threshold is programmable and default to 76 cells

Figure 143

Name	*Type*	*Width*	*Depth*	*Total size*	*Access Time*
Free Buffer List	SSRAM (dual-port)	7 bits	80	800 bits	5ns read/write
SEG_TBL	SSRAM (dual-port)	39 bits	65	2535 bits	5ns read/write
Data SRAM	SSRAM (dual-port)	64 bits	640	40960 bits	5ns read/write
Partial CRC	SSRAM (dual-port)	32 bits	65	2080 bits	5ns read/write
STAT SRAM	SSRAM (single-port)	80 bits	65	5200 bits	5ns read/write
QUE SRAM	SSRAM (single-port)	70 bits	80	5840 bits	5ns read/write

Figure 144

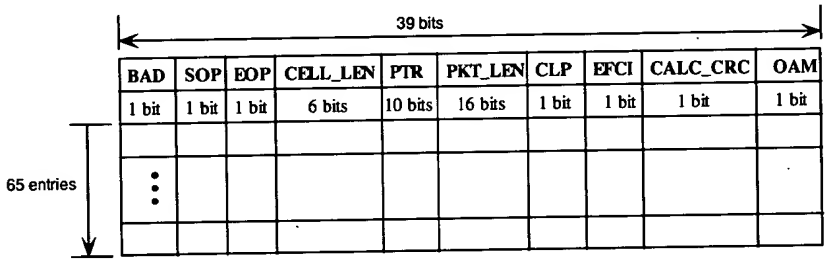


Figure 145

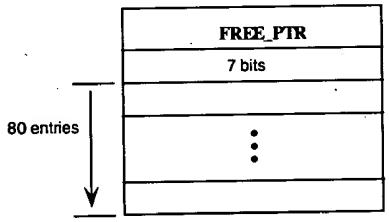


Figure 146

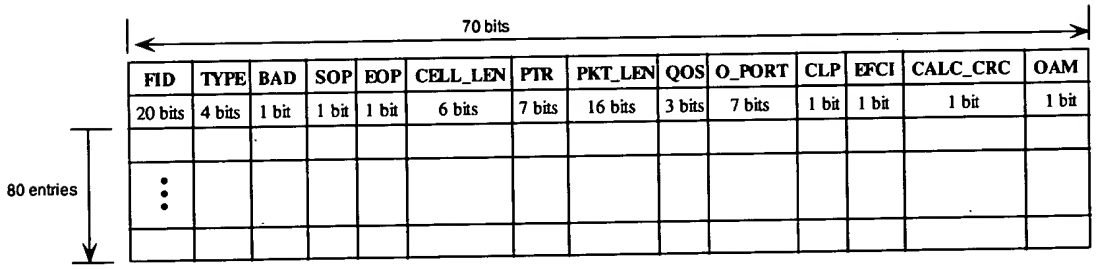


Figure 147

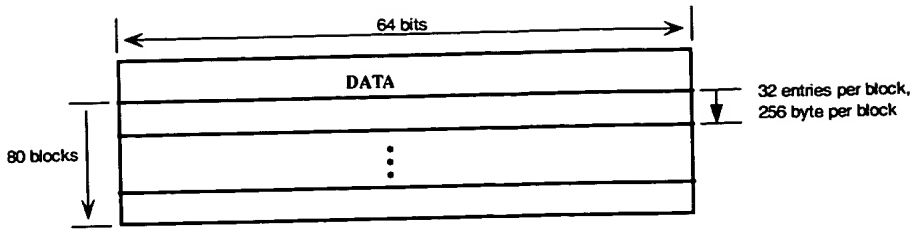


Figure 148

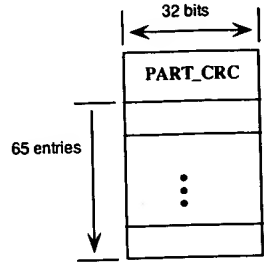


Figure 149

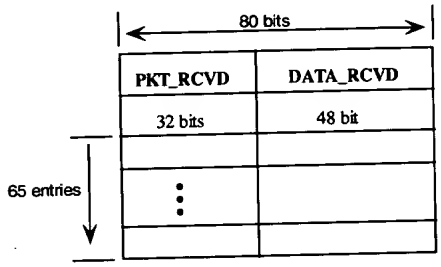


Figure 150

Signal Name	#bits	DIR	Description
SEG_CLK	1	IN	This is the 200 MHz internal system clock to Segmentation block.
G_SYNC	1	IN	
CP_SYNC	1	IN	
CPU_ACCESS	1	IN	

Figure 151

Signal Name	#bits	DIR	Description
LUT_SEG_FID	20	IN	Flow ID
LUT_SEG_PORT	6	IN	Input port number for input data
LUT_SEG_SOB	1	IN	Start of data burst
LUT_SEG_EOB	1	IN	End of data burst
LUT_SEG_SOP	1	IN	Start of Packet
LUT_SEG_EOP	1	IN	End of Packet
LUT_SEG_CLP	1	IN	Cell Loss Priority
LUT_SEG_EFCI	1	IN	EFCI
LUT_SEG_CALC_CRC	1	IN	Calculate L2 CRC
LUT_SEG_CLASS	3	IN	Quality of Service
LUT_SEG_TYPE	4	IN	Type of traffic
LUT_SEG_VALID	1	IN	Port number is valid for sample
LUT_SEG_ERR	1	IN	Discard cell
LUT_SEG_PARITY	1	IN	Parity error indication from SPII
LUT_SEG_OAM	1	IN	Indiacate OAM cell
LUT_SEG_DATA	64	IN	Data in
SEG_LUT_RDY	1	OUT	Segmentation engine ready to receive data from Look up engine

Figure 152

Signal Name	#bits	DIR	Description
MEM_SEG_CTRL_POP	1	IN	Memory Manager pop for data
SEG_MEM_FID	20	OUT	Flow ID to Memory Manager
SEG_MEM_PORT	6	OUT	Output port ID to Memory Manager
SEG_MEM_CLP	1	OUT	CLP bit
SEG_MEM_EFCI	1	OUT	EFCI bit
SEG_MEM_SOP	1	OUT	Start of Packet to Memory Manager
SEG_MEM_EOP	1	OUT	End of Packet to Memory Manager
SEG_MEM_CLASS	3	OUT	Quality of service to Memory Manager
SEG_MEM_TYPE	4	OUT	Types of traffic
SEG_MEM_DISCARD	1	OUT	Indicate packet is to be discarded
SEG_MEM_OAM	1	OUT	Indicate OAM cell
SEG_MEM_AVAILABLE	1	OUT	Cells ready for Memory Manager
SEG_MEM_DATA	64	OUT	Data to Memory Manager

Figure 153

Signal Name	#bits	DIR	Description
PFQ_SEG_CLK	1	IN	Serial clock from PFQ to SEG
PFQ_SEG_VALID	1	IN	Valid signal to start serial transfer
PFQ_SEG_DATA	1	IN	Serial data from PFQ to SEG

Figure 154

Signal Name	#bits	DIR	Description
SEG_SCH_CLK	1	OUT	Serial clock from SEG to SCH
SEG_SCH_VALID	1	OUT	Valid signal to start serial transfer
SEG_SCH_DATA	1	OUT	Serial data from SEG to SCH

Figure 155

Signal Name	#bits	DIR	Description
CPU_SEG_CS_L	1	IN	Block select
CPU_SEG_RDWR_L	1	IN	Block read/write strobe
CPU_SEG_ADDR	6	IN	Block address from CPU
CPU_SEG_DATA_IN	32	IN	Block data from CPU
CPU_SEG_DATA_OUT	32	OUT	Block data from CPU
SEG_RST_L	1	IN	Block reset from CPU
SEG_IN_ENB	1	IN	Block input enable from CPU
SEG_OUT_ENB	1	IN	Block output enable from CPU
SEG_TST_MUX_CLK	1	OUT	Block test clock to CPU
SEG_TST_MUX_OUT	32	OUT	Block test signals to CPU

Figure 156

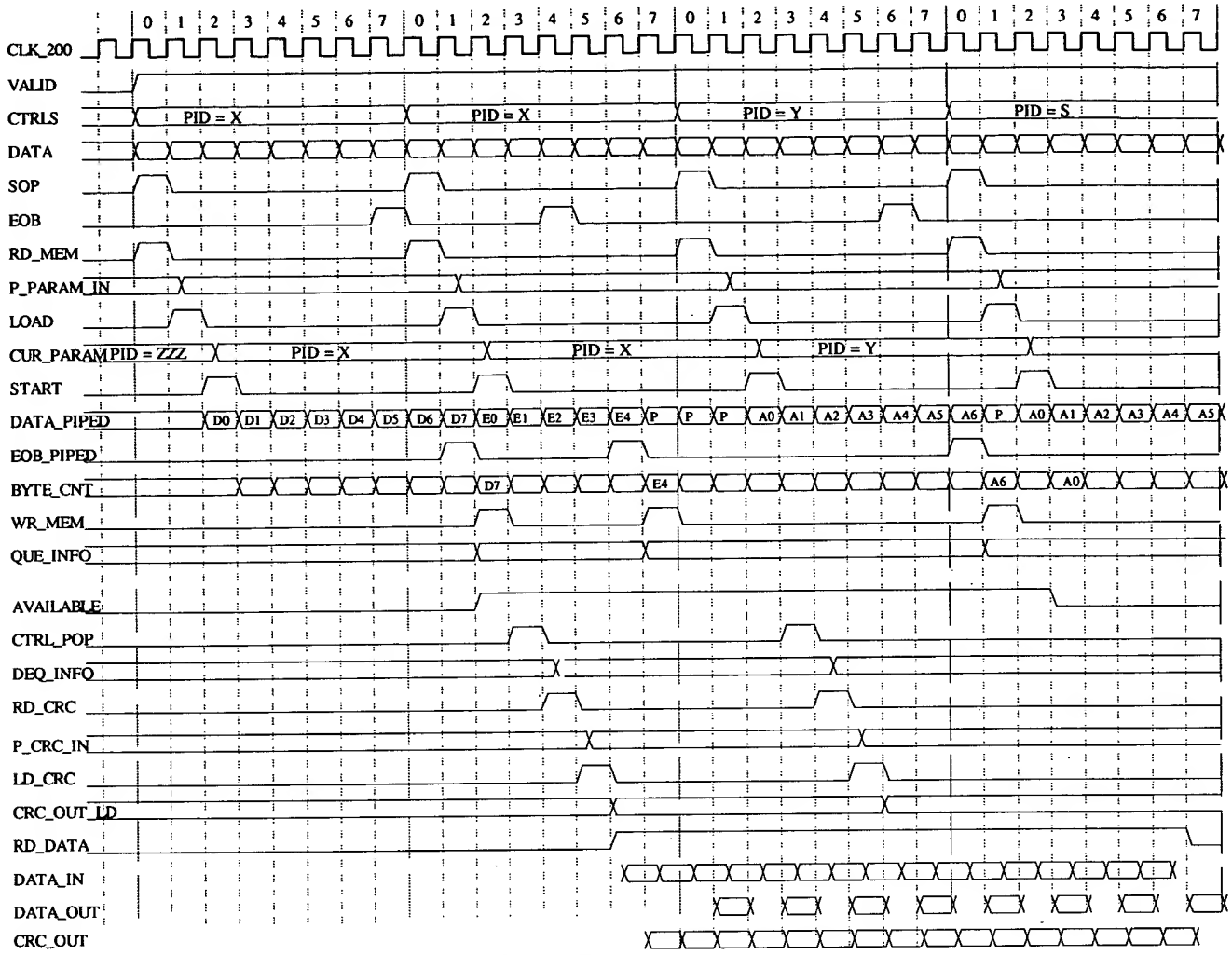


Figure 157

	Cycle0	Cycle1	Cycle2	Cycle3	Cycle4	Cycle5	Cycle6	Cycle7
FBL	Rd	Rd						
TBL	**Rd**		Wr					
STAT	**Rd**		Wr					
DATA	Wr/rd	Wr/rd	Wr/rd	Wr/rd	Wr/rd	Wr/rd	Wr/rd	Wr/rd
QUEUE			Wr	Rd				
CRC						**Rd**		Wr

Figure 158

Address	**Name**	**Type**	**Description**
0	COM	R/W	[31:27] – Opcode [26:0] – Address, depending on the command. No default value.
1	R0	R/W	General-purpose register. No default

			value
2	R1	R/W	General-purpose register. No default value
3	R2	R/W	General-purpose register. No default value
4	R3	R/W	General-purpose register. No default value
5	R4	R/W	General-purpose register. No default value
6	R5	R/W	General-purpose register. No default value
7	R6	R/W	General-purpose register. No default value
8	R7	R/W	General-purpose register. No default value
9	R8	R/W	General-purpose register. No default value
10	R9	R/W	General-purpose register. No default value
11	R10	R/W	General-purpose register. No default value
12	R11	R/W	General-purpose register. No default value
13	R12	R/W	General-purpose register. No default value
14	R13	R/W	General-purpose register. No default value
15	R14	R/W	General-purpose register. No default value
16	R15	R/W	General-purpose register. No default value
17 – 31	Reserved		
32	CPU_CONTROL_0	R/W	[31:20] - Reserved. [19:0] - FID Flow ID of the incoming CPU injected

33	CPU_CONTROL_1	R/W	[31:23] - Reserved [22:16] - O_PORT Output port for the injected data [15:12] - Reserved [11:8] - TYPE Type of traffic/application [7] - Reserved [6:4] - CLASS Quality of Service [3] - Reserved [2] - OAM [1] - EFCI ATM EFCI [0] - CFI Cell Loss Priority, ATM only
34	CPU_PKT_LEN	R/W	[31:16] - Reserved [15:0] - SEG_MTU Maximum transfer unit for packet data
35	SEG_STATUS	R/W	[31:16] - Reserved [0]
36	SEG_MASK	R/W	[19:16] - Reserved [0]
37	SEG_THRESHOLD	R/W	[19:8] – Reserved [7:0] – Programmable FIFO threshold default to 77
38 – 63	Reserved		

Figure 159

OPCODE	Description
00000	IDLE
00001	Init SEG
00010	Read FBL memory
00011	Write FBL memory
00100	Read CRC memory
00101	Write CRC memory
00110	Read STAT memory
00111	Write STAT memory
01000	Read QUEUE memory
01001	Write QUEUE memory
01010	Read DATA memory
01011	Write DATA memory
01100	Read TBL memory
01101	Write TBL memory
01110	CPU inject packet
01111-11111	Reserved

Figure 160

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0	
COM	00001	R[26:2]						SOP	EOP
R0	Cell Data [31:0]								
R1	Cell Data [63:32]								
R2	Cell Data [95:64]								
R3	Cell Data [127:96]								
R4	Cell Data [159:128]								
R5	Cell Data [191:160]								
R6	Cell Data [223:192]								
R7	Cell Data [255:224]								
R8	Cell Data [287:256]								
R9	Cell Data [319:288]								
R10	Cell Data [351:320]								
R11	Cell Data [383:352]								
R12	Cell Data [415:384]								
R13	Cell Data [447:416]								
R14	Cell Data [479:448]								
R15	Cell Data [511:480]								

Figure 161

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00010	R[19:0]					FBL_ADDR[6:0]	
R0	R[21:0]						FBL_PTR[9:0]	

Figure 162

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00011	R[19:0]					FBL_ADDR[6:0]	
R0	R[21:0]						FBL_PTR[9:0]	

FBL_ADDR only addresses from 0 to 79 since there are only 80 free pointers

Figure 163

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00100	R[19:0]					CRC_ADDR[6:0]	
R0	CRC[31:0]							

Figure 164

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00101	R[19:0]					CRC_ADDR[6:0]	
R0	CRC[31:0]							

CRC_ADDR only addresses from 0 to 64.
Addresses 0-63 are the ports and location 64 is for the CPU port.

Figure 165

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00110	R[19:0]					STAT_ADDR[6:0]	
R0	R[15:0]				DATA_CNT[47:32]			
R1	DATA_CNT[31:0]							
R2	PKT_CNT[31:0]							

Figure 166

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00111	R[19:0]					STAT_ADDR[6:0]	
R0	R[15:0]				DATA_CNT[47:32]			
R1	DATA_CNT[31:0]							
R2	PKT_CNT[31:0]							

STAT_ADDR only addresses from 0 to 64.
Addresses 0-63 are the ports and location 64 is for the CPU port.

Figure 167

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01000	R[19:0]					TBL_ADDR[6:0]	
R0	R[11:0]		TYPE[3:0]	S/E/D/C/E/O	O_PORT[6:0]		QOS[2:0]	
R1	PKT_LEN[20:0]				PTR[9:0]		CELL_CNT[5:0]	
R2	R[20:0]			FID[19:0]				

Figure 168

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01001	R[19:0]					TBL_ADDR[6:0]	
R0	R[11:0]			TYPE[3:0]	S/E/D/C/E /O	O_PORT[6:0]		QOS[2:0]
R1	PKT_LEN[20:0]					PTR[9:0]		CELL_CNT[5:0]
R2	R[20:0]				FID[19:0]			

TBL_ADDR only addresses from 0 to 64.
Addresses 0-63 are the ports and location 64 is for the CPU port.

S/E/D/C/E: [14] SOP
[13] EOP
[12] DISCARD
[11] CLP
[10] EFCI
[9] OAM

Figure 169

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01010	R[16:0]				DATA_ADDR[9:0]		
R0	DATA[63:32]							
R1	DATA[31:0]							

Figure 170

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01011	R[16:0]				DATA_ADDR[9:0]		
R0	DATA[63:32]							
R1	DATA[31:0]							

DATA_ADDR only addresses from 0 to 639, 640 entries to store data.

Figure 171

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01100	R[19:0]					TBL_ADDR[6:0]	
R0	R[11:0]			TYPE[3:0]	S/E/D/C/E /O	O_PORT[6:0]		QOS[2:0]
R1	PKT_LEN[15:0]					R[5:0]		PTR[9:0]
R2	R[20:0]				FID[19:0]			

Figure 172

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1101	R[20:0]					TBL_ADDR[6:0]	
R0	R[11:0]		TYPE[4:0]		S/E/D/C/E	O_PORT[6:0]		QOS[2:0]
R1	PKT_LEN[20:0]					PTR[9:0]		CELL_CNT[5:0]
R2	R[20:0]			FID[19:0]				

TBL_ADDR only addresses from 0 to 64.
Addresses 0-63 are the ports and location 64 is for the CPU port.

S/E/D/C/E: [14] SOP
[13] EOP
[12] DISCARD
[11] CLP
[10] EFCI

Figure 173

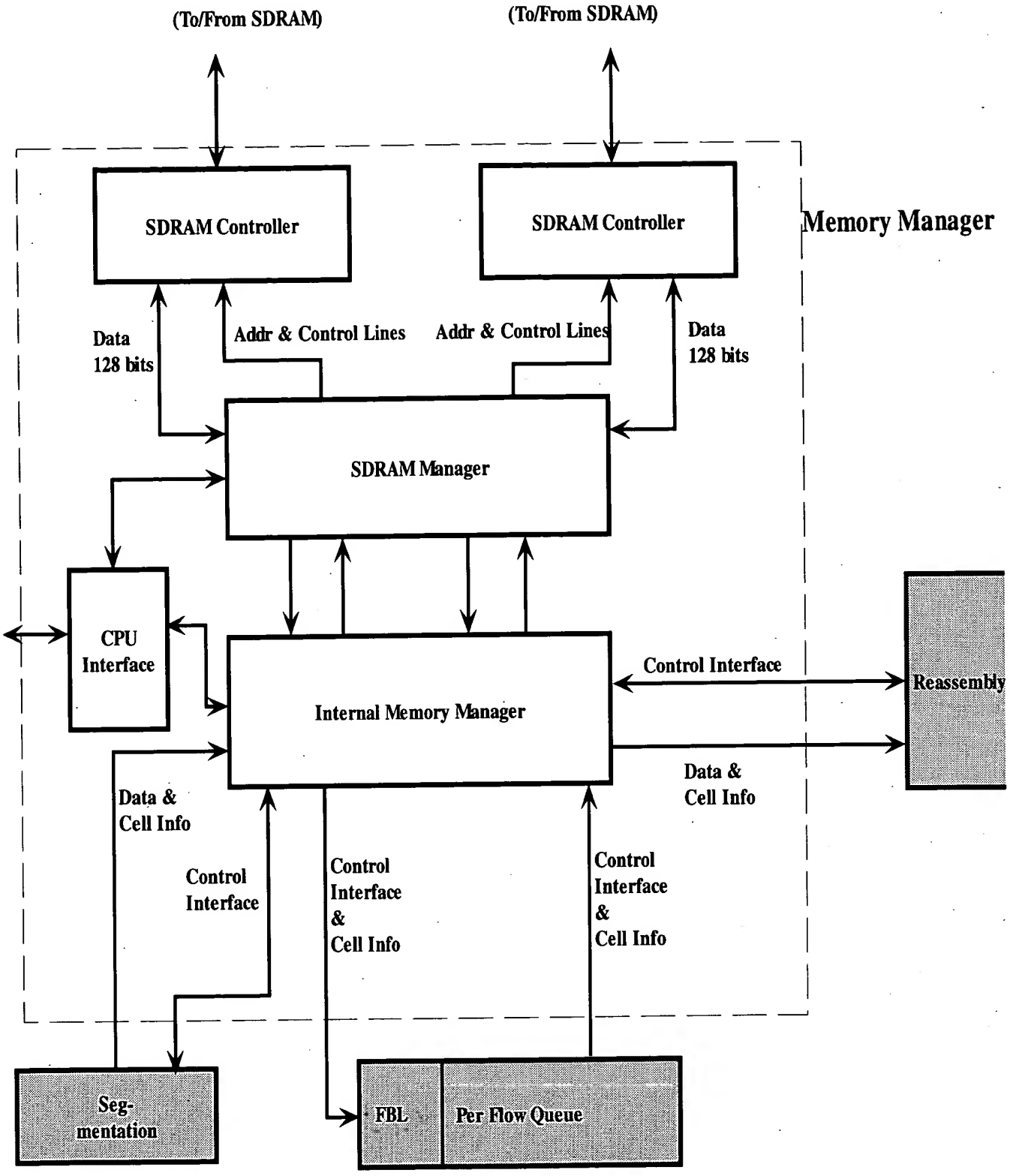

(To/From SDRAM)
(To/From SDRAM)
SDRAM Controller
SDRAM Controller
Memory Manager
Data
128 bits
Addr & Control Lines
Addr & Control Lines
Data
128 bits
SDRAM Manager
CPU
Interface
Internal Memory Manager
Control Interface
Reassembly
Data &
Cell Info
Data &
Cell Info
Control
Interface
Control
Interface
&
Cell Info
Control
Interface
&
Cell Info
Seg-
mentation
FBL
Per Flow Queue

Figure 174

Internal Memory Manager Block Diagram

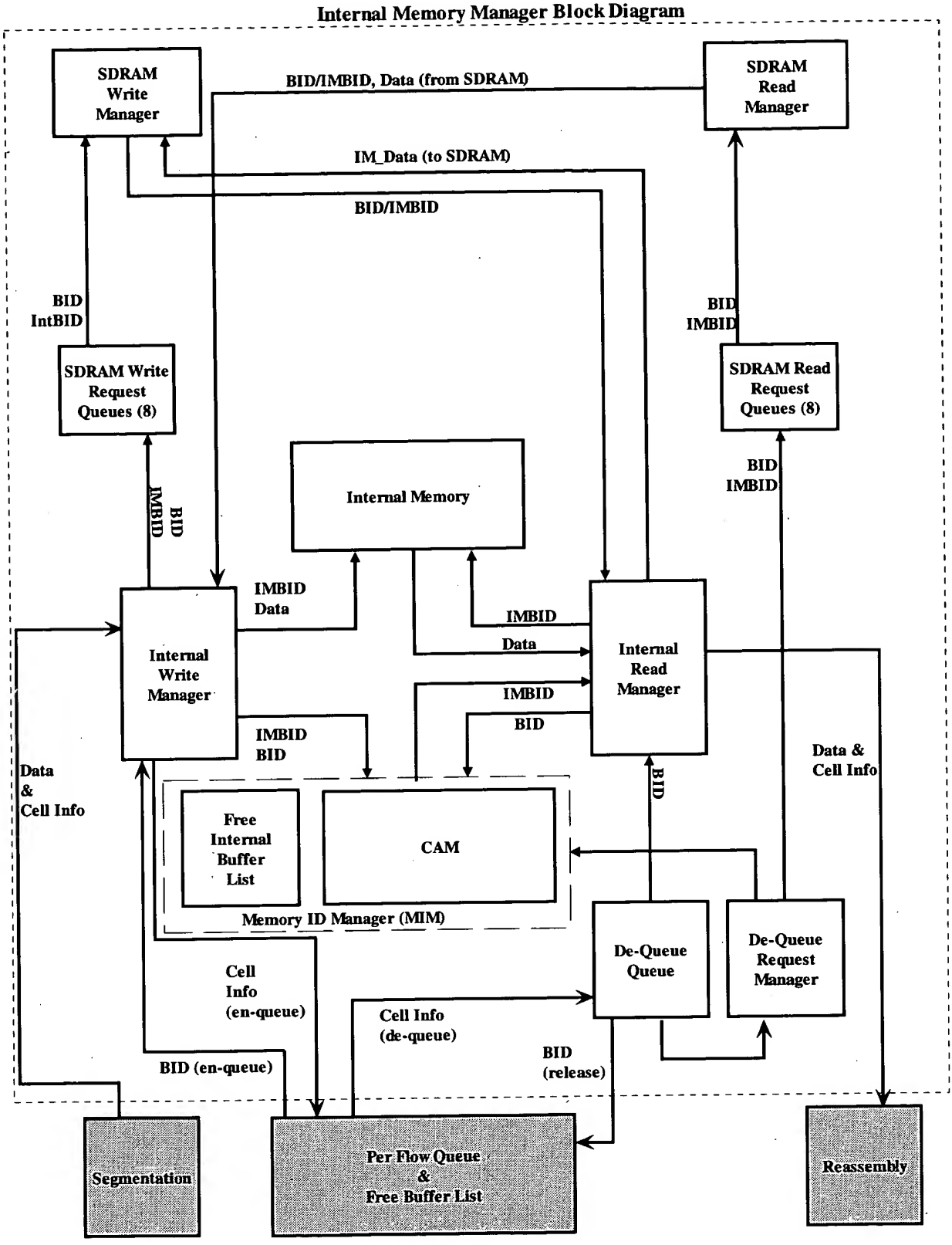


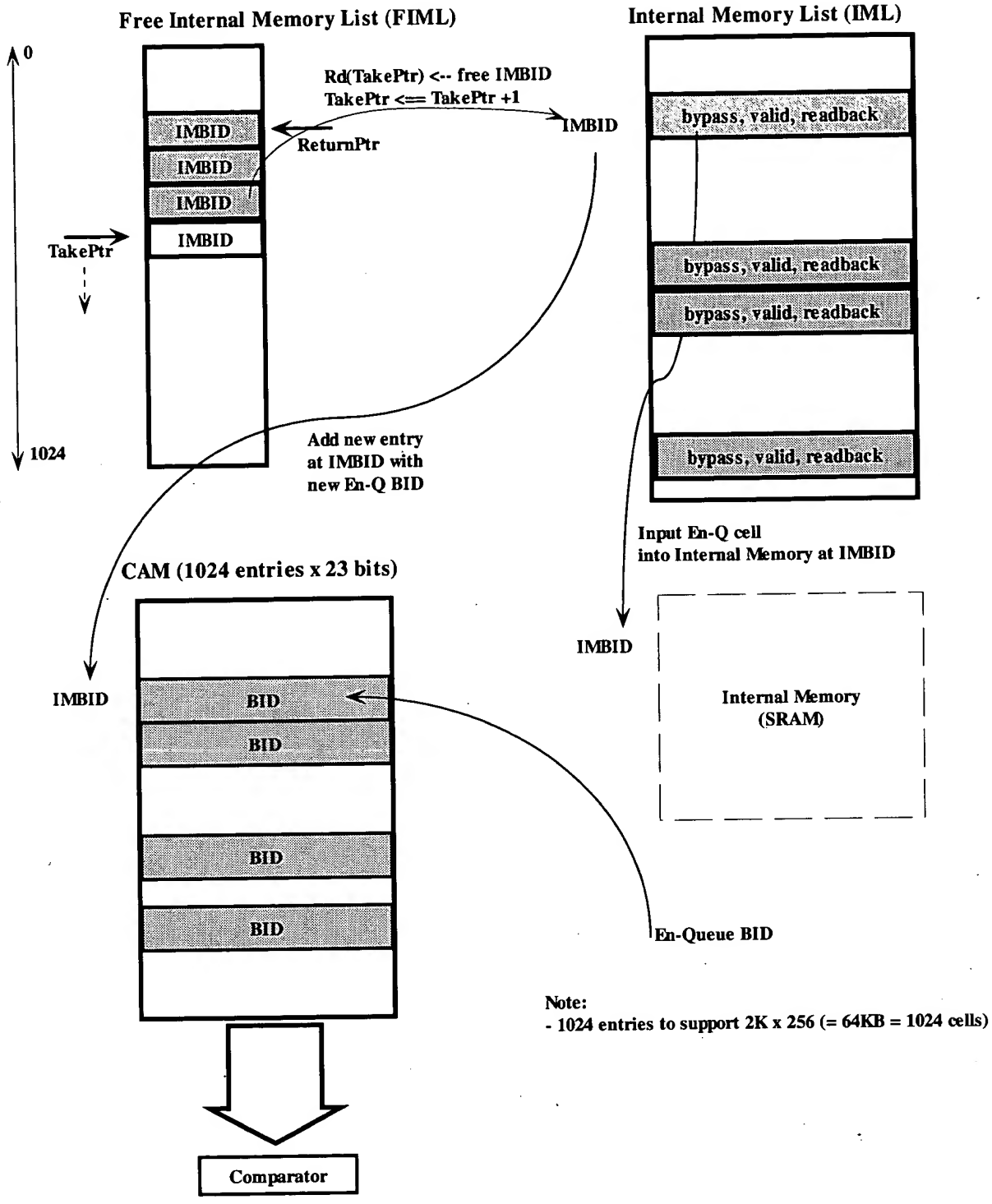
Figure 175

"bypass"	*"read-back"*	*REL =1 BID*	*REL =1 IMBID*	*REL =0 BID*	*REL =0 IMBID*	*Data Type*
0	0	R	R	NR	NR	Non-bypass, has not been transferred to external SDRAM yet
0	1	N/A	N/A	N/A	N/A	N/A
1	0	R	R	NR	NR	Bypass
1	1	R	R	NR	R	Non-bypass, read back from SDRAM

Where:
R: release
NR: NOT RELEASE

Figure 176

EN-QUEUE PROCESS



Note:
- 1024 entries to support 2K x 256 (= 64KB = 1024 cells)

Figure 177

DE-QUEUE PROCESS

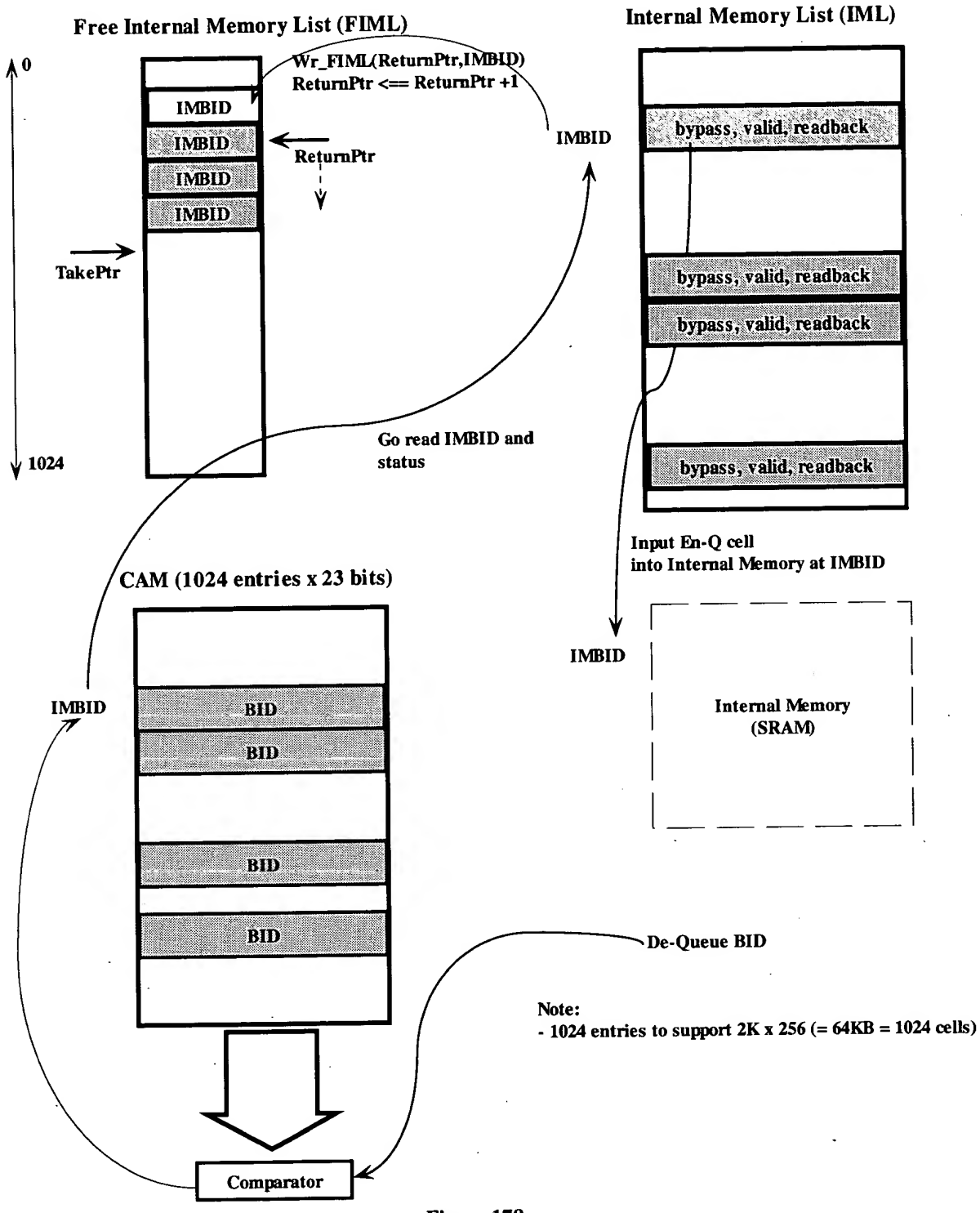


Figure 178

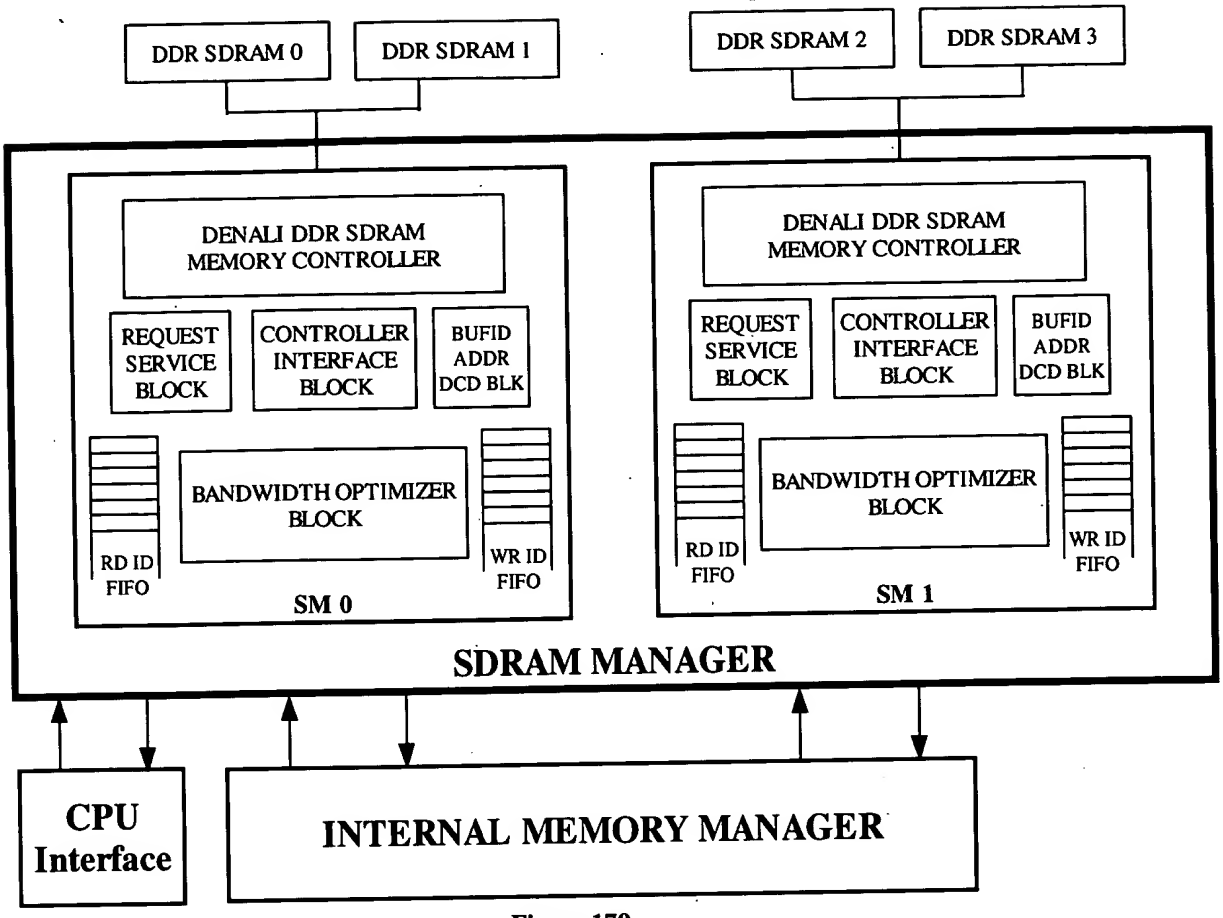


Figure 179

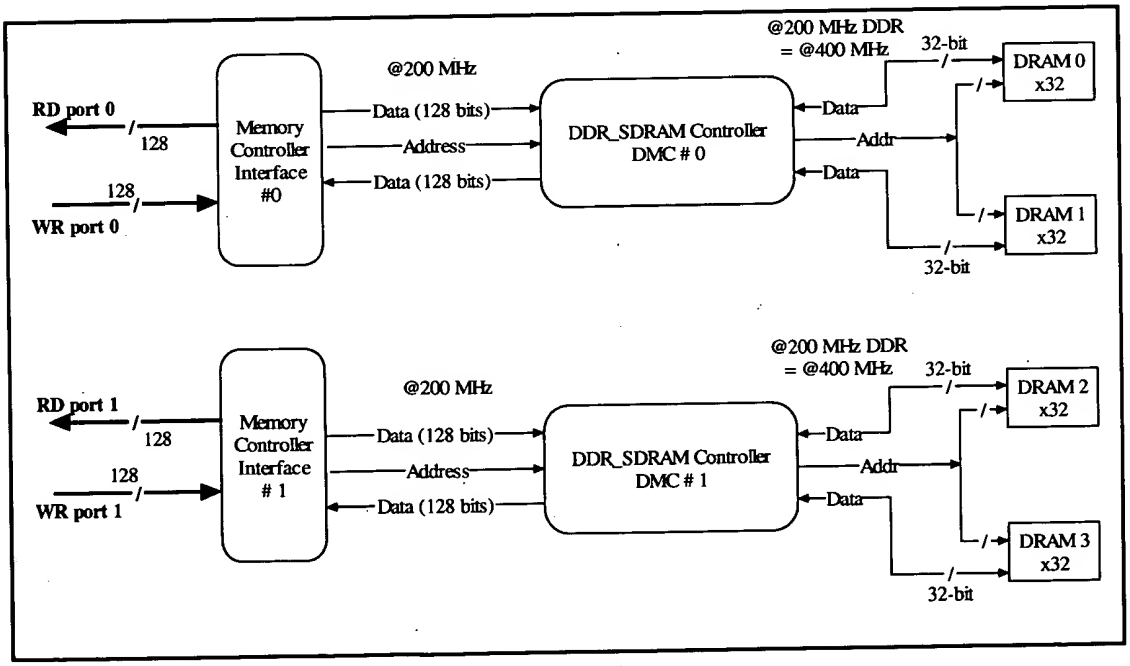


Figure 180

Name	*Type*	*Width*	*Depth*	*Total size*	*Access Time*
Internal Memory (IM)	SSRAM (dual port)	128 bits	4K	1Mbits	5ns read/write
Internal Memory List (IML)	Flip-flops	3	1K	3072 bits	5ns read/write
Free Internal Memory List (FIML)	SSRAM (single port)	10 bits	1K	10Kbits	5ns read/write
Internal Write Manager FIFO (IWMFIFO)	SSRAM (dual port)	128 bits	16	2Kbits	5ns read/write

Figure 181

Name	*Type*	*Width*	*Depth*	*Total size*	*Access Time*
CAM (inside MIM)	CAM	23	1K	23Kbits	

Figure 182

Signal Names	Size	Dir	Description
Seg_Mem_Data	128 bits	In	Input data from Segmentation at 10 ns period
Mem_Seg_Cntl_Pop	1	Out	To pop control information and data of a cell
Seg_Mem_Available	1	In	Data available in Segmentation for MM
Seg_Mem_FID	20	In	Flow ID
Seg_Mem_EOP	1	In	End of Packet
Seg_Mem_SOP	1	In	Start of Packet
Seg_Mem_Discard	1	In	Signal cell to be discard
Seg_Mem_Type	5	In	Type
Seg_Mem_O_Port	7	In	Output port information
Seg_Mem_Class	3	In	Class information
Seg_Mem_EFCI	1	In	EFCI
Seg_Mem_CLP	1	In	CLP
Seg_Mem_OAM	1	In	

Figure 183

Signal Names	Size	Dir	Description
Mem_PFQ_Clk	1	Out	Clock from Mem to PFQ for synchronization
Mem_PFQ_Com	2	Out	Request command to PFQ: 00: not valid 01: enqueue 10: discard 11: dequeue release BID
Mem_PFQ_Valid	1	Out	Signal Mem_PFQ_Data is valid
Mem_PFQ_Data	24	Out	Cell control information from MM to PFQ for both enqueue and dequeue
Mem_PFQ_Full	1	Out	To indicate De-queue queues almost full to PFQ to stop de-queuing request
PFQ_Mem_Clk	1	In	Clock from PFQ to Mem for synchronization
PFQ_Mem_Com	2	In	Request command to Mem: 00: no command 10: invalid command 10: dequeue current cell 11: invalid command
PFQ_Mem_Valid	1	In	Signal Mem_PFQ_Data is valid
PFQ_Mem_Data	24	In	Cell control information from PFQ to MM for both enqueue and dequeue

Figure 184

Signal Names	Size	Dir	Description
Mem_SDR_Data_A	64	In/Out	Data bus for Controller A
Mem_SDR_DQS_Lo_A	1	In/Out	Data strobe of lower 32bit of Data for Controller A (one line per SDRAM)
Mem_SDR_DQS_Hi_A	1	In/Out	Data strobe of upper 32bit of Data for Controller A (one line per SDRAM)
Mem_SDR_CS_A	8	Out	Chip selects to support up to 512 Mbytes
Mem_SDR_Addr0_A	12	Out	Row/column address bus for Controller A (set 0)
Mem_SDR_Addr1_A	12	Out	Row/column address bus for Controller A (set 1, duplication of Mem_SDR_Addr0_A)
Mem_SDR_Bank0_A	2	Out	Bank selection for Controller A (set 0)
Mem_SDR_Bank1_A	2	Out	Bank selection for Controller A (set 1)
Mem_SDR_RAS0_A	1	Out	RAS for Controller A (set 0)
Mem_SDR_RAS1_A	1	Out	RAS for Controller A (set 1)
Mem_SDR_CAS0_A	1	Out	CAS for Controller A (set 0)
Mem_SDR_CAS1_A	1	Out	CAS for Controller A (set 1)
Mem_SDR_WE0_A	1	Out	Write enable for Controller A (set 0)
Mem_SDR_WE1_A	1	Out	Write enable for Controller A (set 1)
Mem_SDR_Clk0_A	1	Out	Differential clock output for Controller A

			(set 0)
Mem_SDR_Clk0#_A	1	Out	Differential clock output for Controller A (set 0)
Mem_SDR_Clk1_A	1	Out	Differential clock output for Controller A (set 1)
Mem_SDR_Clk1#_A	1	Out	Differential clock output for Controller A (set 1)
Mem_SDR_Clke_A	1	Out	Clock enable for Controller A
Mem_SDR_Data_B	64	In/Out	Data bus for Controller B
Mem_SDR_DQS_Lo_B	1	In/Out	Data strobe of lower 32bit of Data for Controller B (one line per SDRAM)
Mem_SDR_DQS_Hi_B	1	In/Out	Data strobe of upper 32bit of Data for Controller B (one line per SDRAM)
Mem_SDR_CS_B	8	Out	Chip selects to support up to 512 Mbytes
Mem_SDR_Addr0_B	12	Out	Row/column address bus for Controller B (set 0)
Mem_SDR_Addr1_B	12	Out	Row/column address bus for Controller B (set 1, duplication of Mem_SDR_Addr0_B)
Mem_SDR_Bank0_B	2	Out	Bank selection for Controller B (set 0)
Mem_SDR_Bank1_B	2	Out	Bank selection for Controller B (set 1)
Mem_SDR_RAS0_B	1	Out	RAS for Controller B (set 0)
Mem_SDR_RAS1_B	1	Out	RAS for Controller B (set 1)
Mem_SDR_CAS0_B	1	Out	CAS for Controller B (set 0)
Mem_SDR_CAS1_B	1	Out	CAS for Controller B (set 1)
Mem_SDR_WE0_B	1	Out	Write enable for Controller B (set 0)
Mem_SDR_WE1_B	1	Out	Write enable for Controller B (set 1)
Mem_SDR_Clk0_B	1	Out	Differential clock output for Controller B (set 0)
Mem_SDR_Clk0#_B	1	Out	Differential clock output for Controller B (set 0)
Mem_SDR_Clk1_B	1	Out	Differential clock output for Controller B (set 1)
Mem_SDR_Clk1#_B	1	Out	Differential clock output for Controller B (set 1)
Mem_SDR_Clke_B	1	Out	Clock enable for Controller B

Figure 185

Signal Names	Size	Dir	Description
Mem_Ras_Data	128	Out	Output data to Reassembly at 10 ns period
Mem_Ras_Available	1	Out	Data is available in Read Fifo
Ras_Mem_Control_Pop	1	In	Pop signal from Reassembly to pop control info
Ras_Mem_Data_Pop	1	In	Pop signal from Reassembly to pop data
Mem_Ras_Type	5	Out	Cell type
Mem_Ras_FID	20	Out	Flow ID
Mem_Ras_EOP	1	Out	End of Packet
Mem_Ras_SOP	1	Out	Start of Packet
Mem_Ras_PortID	7	Out	Port ID
Mem_Ras_EFCI	1	Out	EFCI
Mem_Ras_CLP	1	Out	CLP

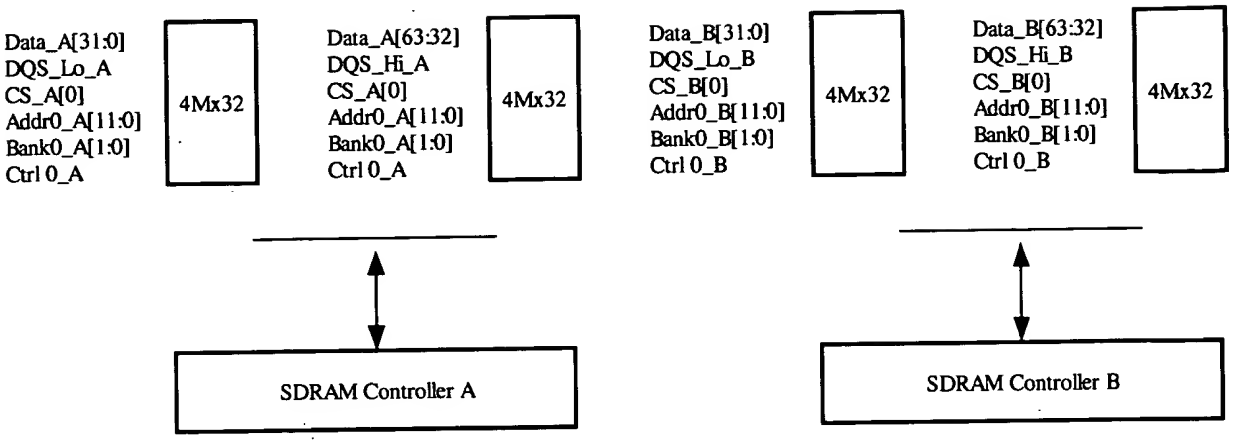
Figure 186

Signal Names	Size	Dir	Description
CPU_Mem_Data_Out	32	Out	Data Out to CPU
CPU_Data_In	32	In	CPU Data In
CPU_Addr	6	In	Register Address
CPU_RDWR_L	1	In	CPU Read/Write Signal
CPU_Mem_CS_L	1	In	MM Block Select
Mem_Reset_L	1	In	MM Reset from CPU
Mem_Input_Ena	1	In	MM Input Enable from CPU
Mem_Output_Ena	1	In	MM Output Enable from CPU
Mem_Tstmux_Out	32	Out	MM Test Signals Out to Test Mux

Figure 187

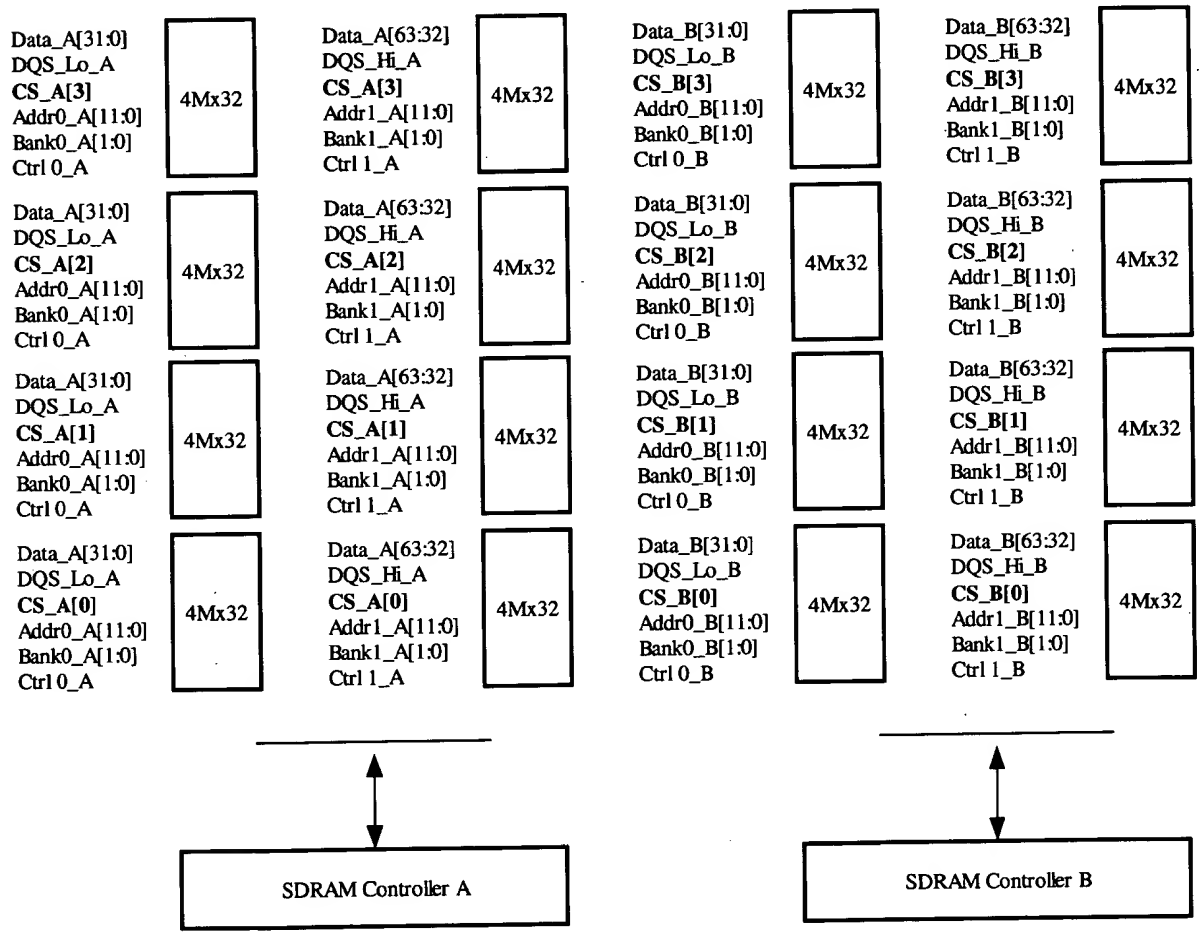
Signal Names	Size	Dir	Description
GS_Mem_Gsync	1	In	Sync pulse from Global Sync block

Figure 188



** Ctrl 0_A, Ctrl 0_B: Control Sigals like RAS_, CAS_, WE_, CLK, CLK_for Controller A and B respectively*

Figure 189



** Ctrl 0_A, Ctrl 0_B: Set 0 of Control Sigals like RAS_, CAS_, WE_, CLK, CLK_for Controller A and B respectively*
** Ctrl 1_A, Ctrl 1_B: Set 1 of Control Sigals like RAS_, CAS_, WE_, CLK, CLK_for Controller A and B respectively*

Figure 190

Address	Name	Type	Description
0	COM	R/W	[31:27] – Opcode [26:0] – Address, depending on the command. No default value.
1	R0	R/W	General-purpose register. No default value
2	R1	R/W	General-purpose register. No default value
3	R2	R/W	General-purpose register. No default value
4	R3	R/W	General-purpose register. No default value
5	R4	R/W	General-purpose register. No default value
6	R5	R/W	General-purpose register. No default value
7	R6	R/W	General-purpose register. No default value
8	R7	R/W	General-purpose register. No default value
9	R8	R/W	General-purpose register. No default value
10	R9	R/W	General-purpose register. No default value
11	R10	R/W	General-purpose register. No default value
12	R11	R/W	General-purpose register. No default value
13	R12	R/W	General-purpose register. No default value
14	R13	R/W	General-purpose register. No default value
15	R14	R/W	General-purpose register. No default value
16	R15	R/W	General-purpose register. No default value
17 – 31	Reserved		
32	MEM_CONFIG	R/W	[2:0] – Memory_Size: 000: 64Mbytes = 1Mcells (default) 001: 128Mbytes = 2Mcells 010: 192Mbytes = 3Mcells 011: 256Mbytes = 4Mcells 100: 320Mbytes = 5Mcells 101: 384Mbytes = 6Mcells 110: 448Mbytes = 7Mcells 111: 512Mbytes = 8Mcells [3] – Enable_Second_set of SDRAM Address and Control signals (for improving signal driving capability [31:4] – Reserved
36	MEM_STATUS_0	R/W	[0] – IM_Full: Internal Memory is full. [1] – IM_Full_1: Internal Memory is Full – 1 cell. [2] – IM_Empty: Internal Memory is empty. [3] – Dequeue_Queue_Full: Dequeue Queue is full. [4] – Dequeue_Queue_Empty: Dequeue Queue is empty. [31:5] - Reserved

37	MEM_STATUS_1	R/W	[0] – SRRQ_0_Full: SDRAM Read Request queue 0 is full. [1] – SRRQ_0_Empty: SDRAM Read Request queue 0 is empty. [2] – SRRQ_1_Full: SDRAM Read Request queue 1 is full. [3] – SRRQ_1_Empty: SDRAM Read Request queue 1 is empty. [4] – SRRQ_2_Full: SDRAM Read Request queue 2 is full. [5] – SRRQ_2_Empty: SDRAM Read Request queue 2 is empty. [6] – SRRQ_3_Full: SDRAM Read Request queue 3 is full. [7] – SRRQ_3_Empty: SDRAM Read Request queue 3 is empty. [8] – SRRQ_4_Full: SDRAM Read Request queue 4 is full. [9] – SRRQ_4_Empty: SDRAM Read Request queue 4 is empty. [10] – SRRQ_5_Full: SDRAM Read Request queue 5 is full. [11] – SRRQ_5_Empty: SDRAM Read Request queue 5 is empty. [12] – SRRQ_6_Full: SDRAM Read Request queue 3 is full. [13] – SRRQ_6_Empty: SDRAM Read Request queue 6 is empty. [14] – SRRQ_7_Full: SDRAM Read Request queue 7 is full. [15] – SRRQ_7_Empty: SDRAM Read Request queue 7 is empty. [16] – SWRQ_0_Full: SDRAM Write Request queue 0 is full. [17] – SWRQ_0_Empty: SDRAM Write Request queue 0 is empty. [18] – SWRQ_1_Full: SDRAM Write Request queue 1 is full. [19] – SWRQ_1_Empty: SDRAM Write Request queue 1 is empty. [20] – SWRQ_2_Full: SDRAM Write Request queue 2 is full. [21] – SWRQ_2_Empty: SDRAM Write Request queue 2 is empty. [22] – SWRQ_3_Full: SDRAM Write Request queue 3 is full. [23] – SWRQ_3_Empty: SDRAM Write Request queue 3 is empty. [24] – SWRQ_4_Full: SDRAM Write Request queue 4 is full. [25] – SWRQ_4_Empty: SDRAM Write Request queue 4 is empty. [26] – SWRQ_5_Full: SDRAM Write Request queue 5 is full. [27] – SWRQ_5_Empty: SDRAM Write Request queue 5 is empty. [28] – SWRQ_6_Full: SDRAM Write Request queue 6 is full. [29] – SWRQ_6_Empty: SDRAM Write Request queue 6 is empty. [30] – SWRQ_7_Full: SDRAM Write Request queue 7 is full. [31] – SWRQ_7_Empty: SDRAM Write Request queue 2 is empty.
38	MEM_TSTMUX_SEL	RW	[3:0] – Testmux_Selection: 0000: No output (default) 0001: Group 1 = {TBD} 0010: Group 2 = {TBD} 0011: Group 3 = {TBD} 0100: Group 4 = {TBD} 0101: Group 5 = {TBD} 0110: Group 6 = {TBD} 0111: Group 7 = {TBD} 1000: Group 8 = {TBD} 1001: Group 9 = {TBD} 1010: Group 10 = {TBD} 1011: Group 11 = {TBD} 1100: Group 12 = {TBD} 1101: Group 13 = {TBD} 1110: Group 14 = {TBD} 1111: Group 15 = {TBD}
39 - 63	Reserved		

Figure 191

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00001	R[26:23]	CELL_ADDR[22:0]					
R0	Cell Data [31:0]							
R1	Cell Data [63:32]							
R2	Cell Data [95:64]							
R3	Cell Data [127:96]							
R4	Cell Data [159:128]							
R5	Cell Data [191:160]							
R6	Cell Data [223:192]							
R7	Cell Data [255:224]							
R8	Cell Data [287:256]							
R9	Cell Data [319:288]							
R10	Cell Data [351:320]							
R11	Cell Data [383:352]							
R12	Cell Data [415:384]							
R13	Cell Data [447:416]							
R14	Cell Data [479:448]							
R15	Cell Data [511:480]							

Figure 192

egister	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00010	R[26:23]	CELL_ADDR[22:0]					
R0	Cell Data [31:0]							
R1	Cell Data [63:32]							
R2	Cell Data [95:64]							
R3	Cell Data [127:96]							
R4	Cell Data [159:128]							
R5	Cell Data [191:160]							
R6	Cell Data [223:192]							
R7	Cell Data [255:224]							
R8	Cell Data [287:256]							
R9	Cell Data [319:288]							
R10	Cell Data [351:320]							
R11	Cell Data [383:352]							
R12	Cell Data [415:384]							
R13	Cell Data [447:416]							
R14	Cell Data [479:448]							
R15	Cell Data [511:480]							

Figure 193

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00011	R[26:10]					FIML_ADDR[9:0]	

Figure 194

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00100	R[26:10]					FIML_ADDR[9:0]	
R0	R[27:10]						FIML[9:0]	

Figure 195

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00101	R[26:10]					CAM_ADDR[9:0]	
R0		IML[25:23]		CAM[22:0]				

Figure 196

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00110	R[26:10]					CAM_ADDR[9:0]	
R0		IML[25:23]		CAM[22:0]				

Figure 197

Register	31-27	26-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00111	R[26:0]						

Figure 198

Inputs from Segmentation

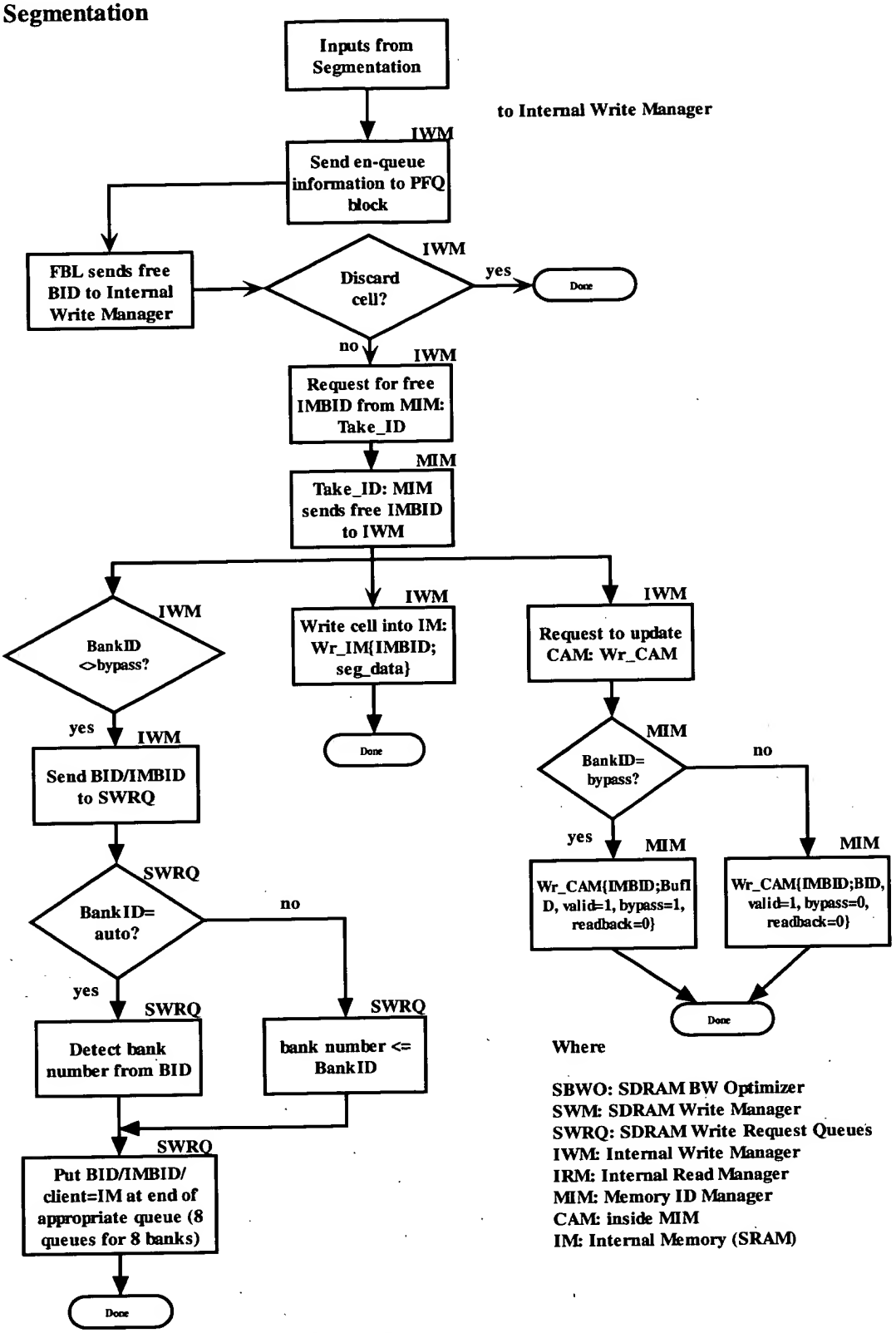
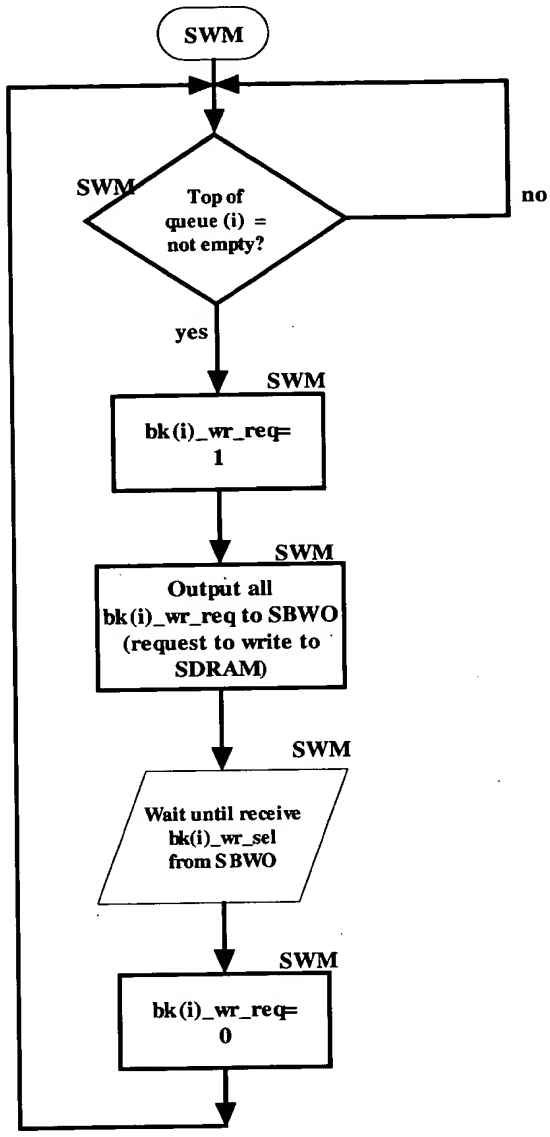


Figure 199

Interface with SDRAM BW Optimizer: SWM requests to Write to SDRAM



Note:

There are 8 queues (i: 0...7) inside SWRQ which associated with 8 banks of SDRAM.

Where
i = 0 ... 7
SBWO: SDRAM BW Optimizer
SWM: SDRAM Write Manager
SWRQ: SDRAM Write Request Queues
IWM: Internal Write Manager
IRM: Internal Read Manager
MIM: Memory ID Manager
CAM: inside MIM
IM: Internal Memory (SRAM)

Figure 200

Interface with SDRAM BW Optimizer: Write to SDRAM

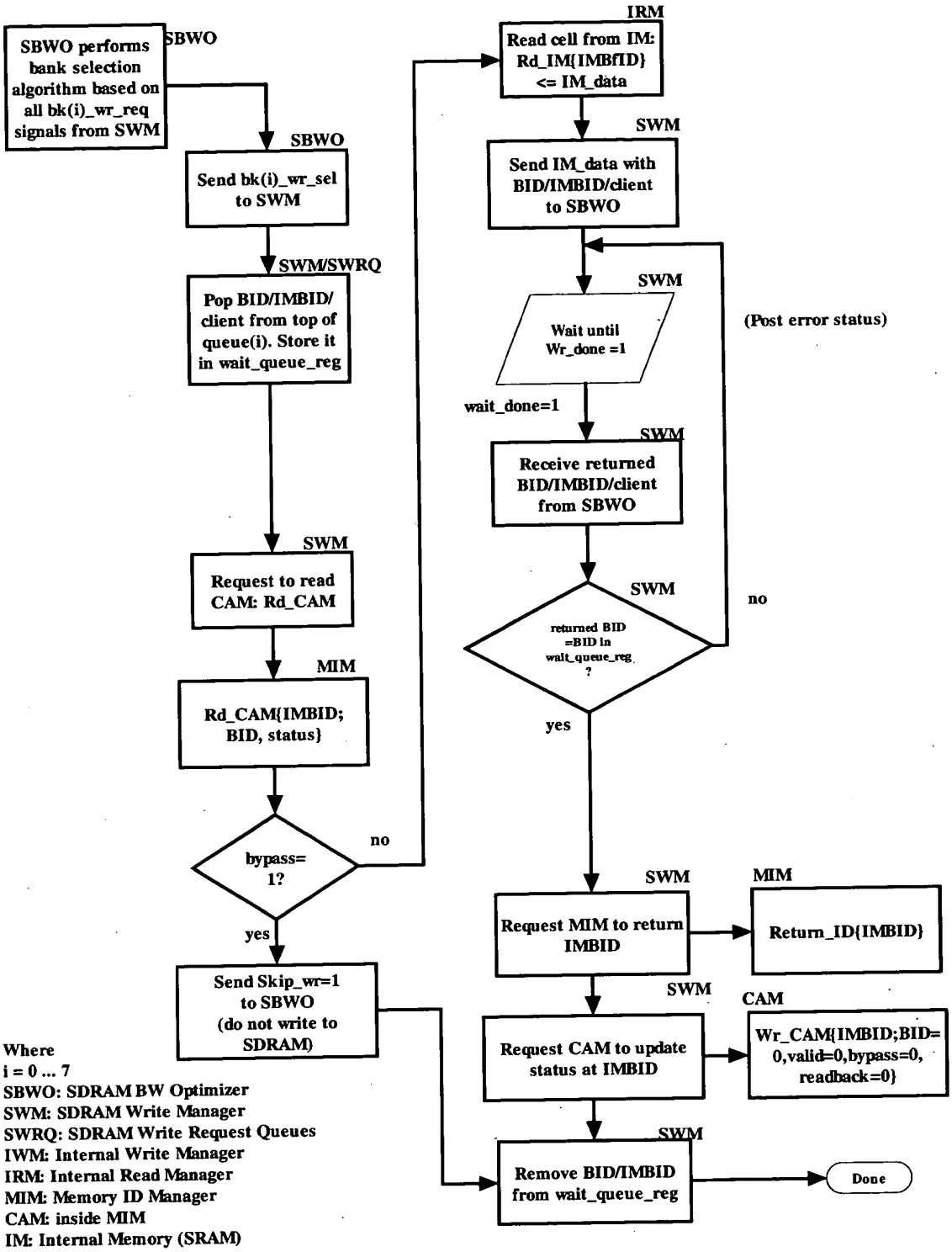
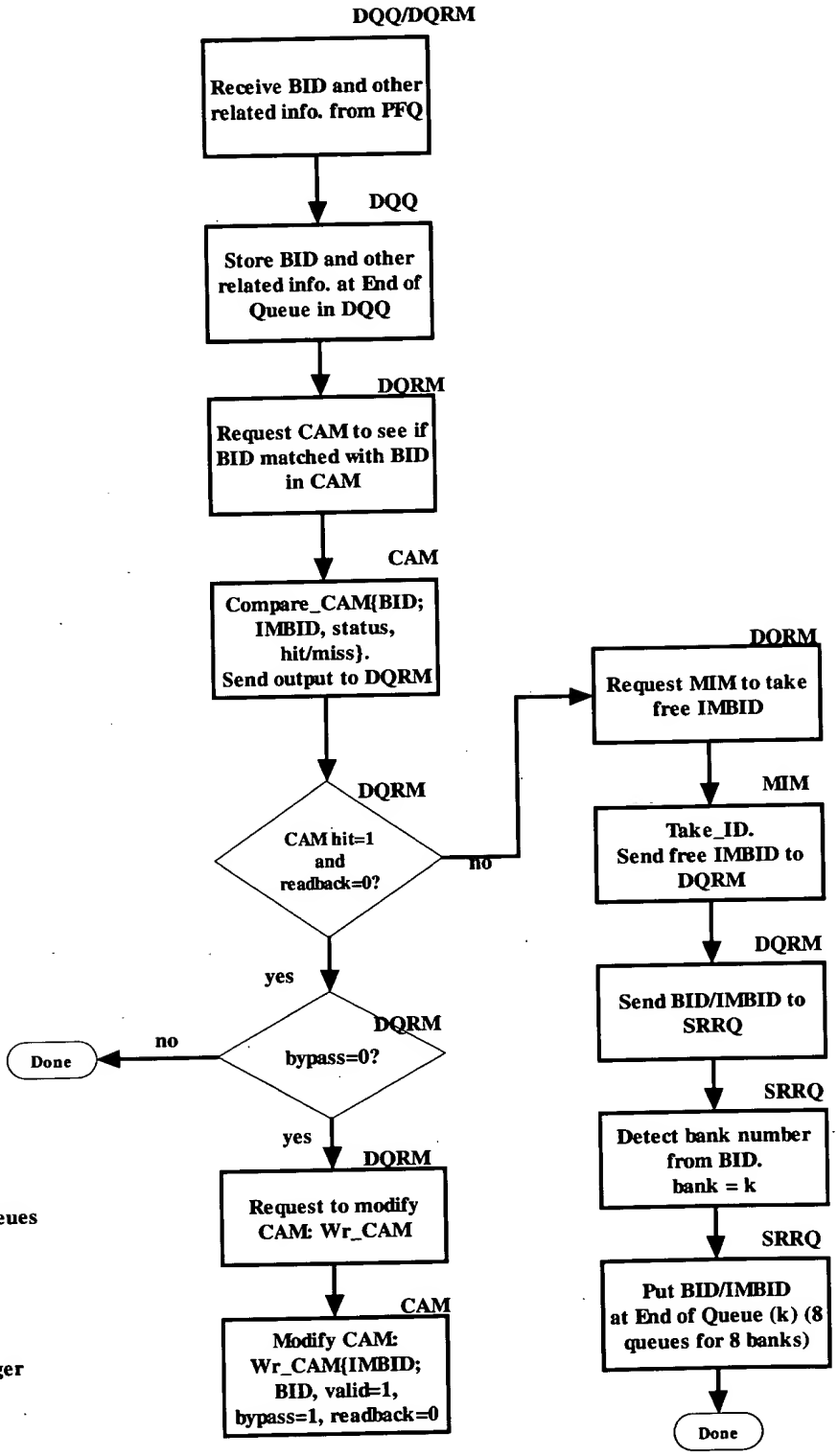


Figure 201

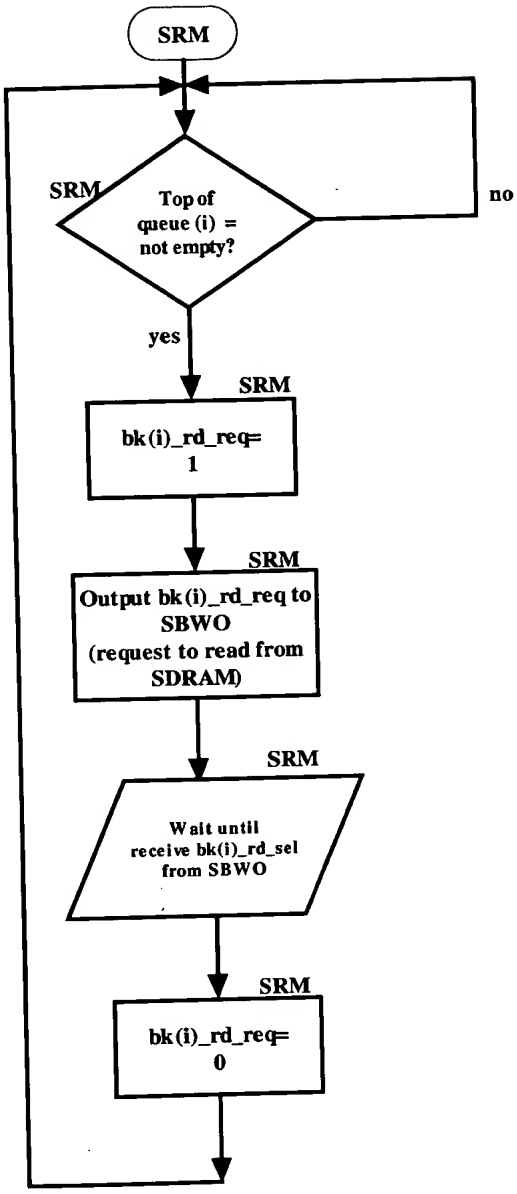
Interface with Per Flow Queue:
Receive De-Queue Requests from PFQ



Where
k = 0 ... 7
SBWO: SDRAM BW Optimizer
SRM: SDRAM Write Manager
SRRQ: SDRAM Read Request Queues
IWM: Internal Write Manager
IRM: Internal Read Manager
MIM: Memory ID Manager
CAM: inside MIM
IM: Internal Memory (SRAM)
DQRM: De-Queue Request Manager
DQQ: De-Queue Queue

Figure 202

Interface with SDRAM BW Optimizer: SRM requests to Read from SDRAM



Note:

There are 8 queues (i: 0...7) inside SWRQ which associated with 8 banks of SDRAM.

Where
i = 0 ... 7
SBWO: SDRAM BW Optimizer
SWM: SDRAM Write Manager
SWRQ: SDRAM Write Request Queues
IWM: Internal Write Manager
IRM: Internal Read Manager
MIM: Memory ID Manager
CAM: inside MIM
IM: Internal Memory (SRAM)

Figure 203

Interface with SDRAM BW Optimizer:
SRM Read from SDRAM

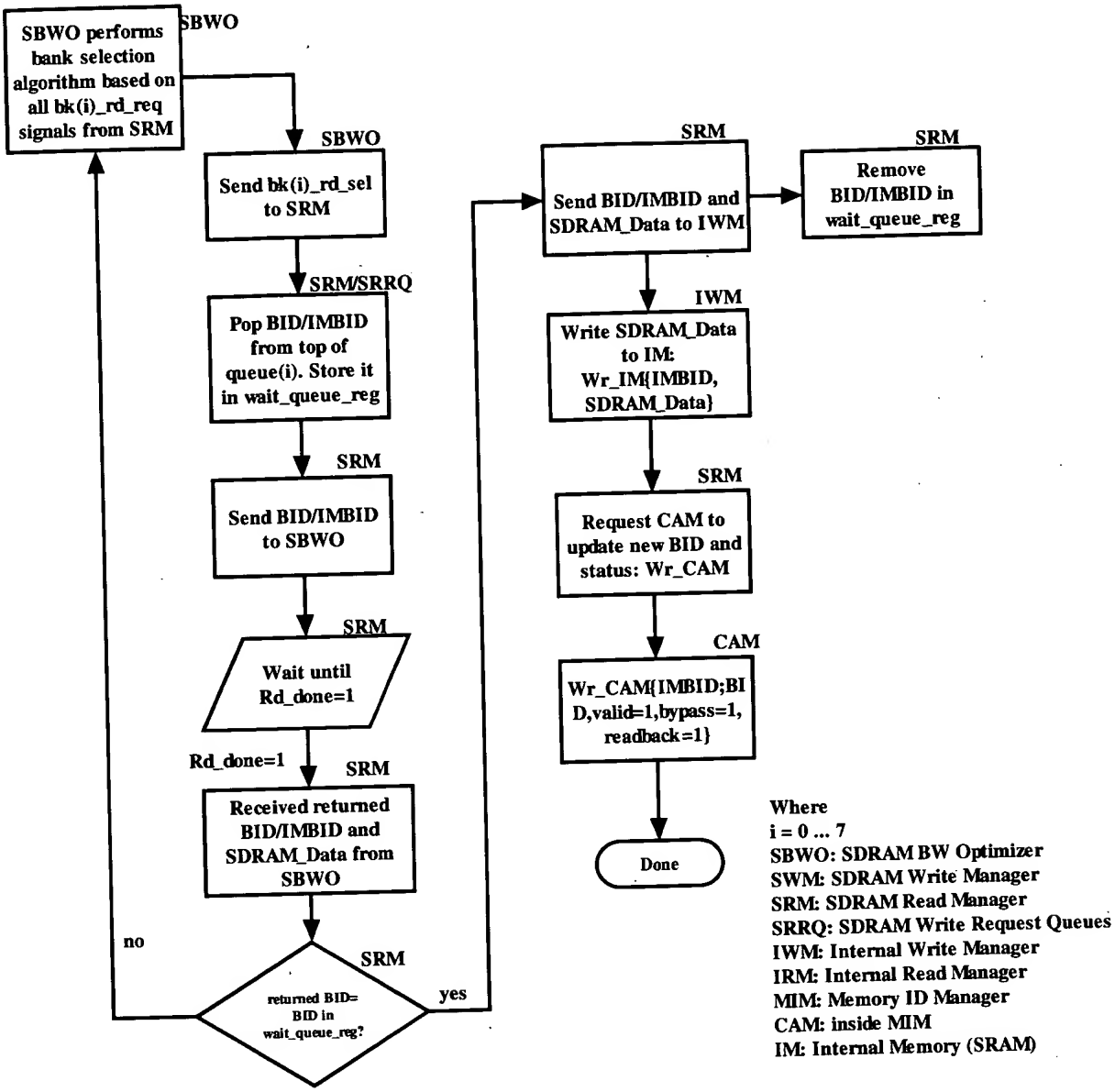
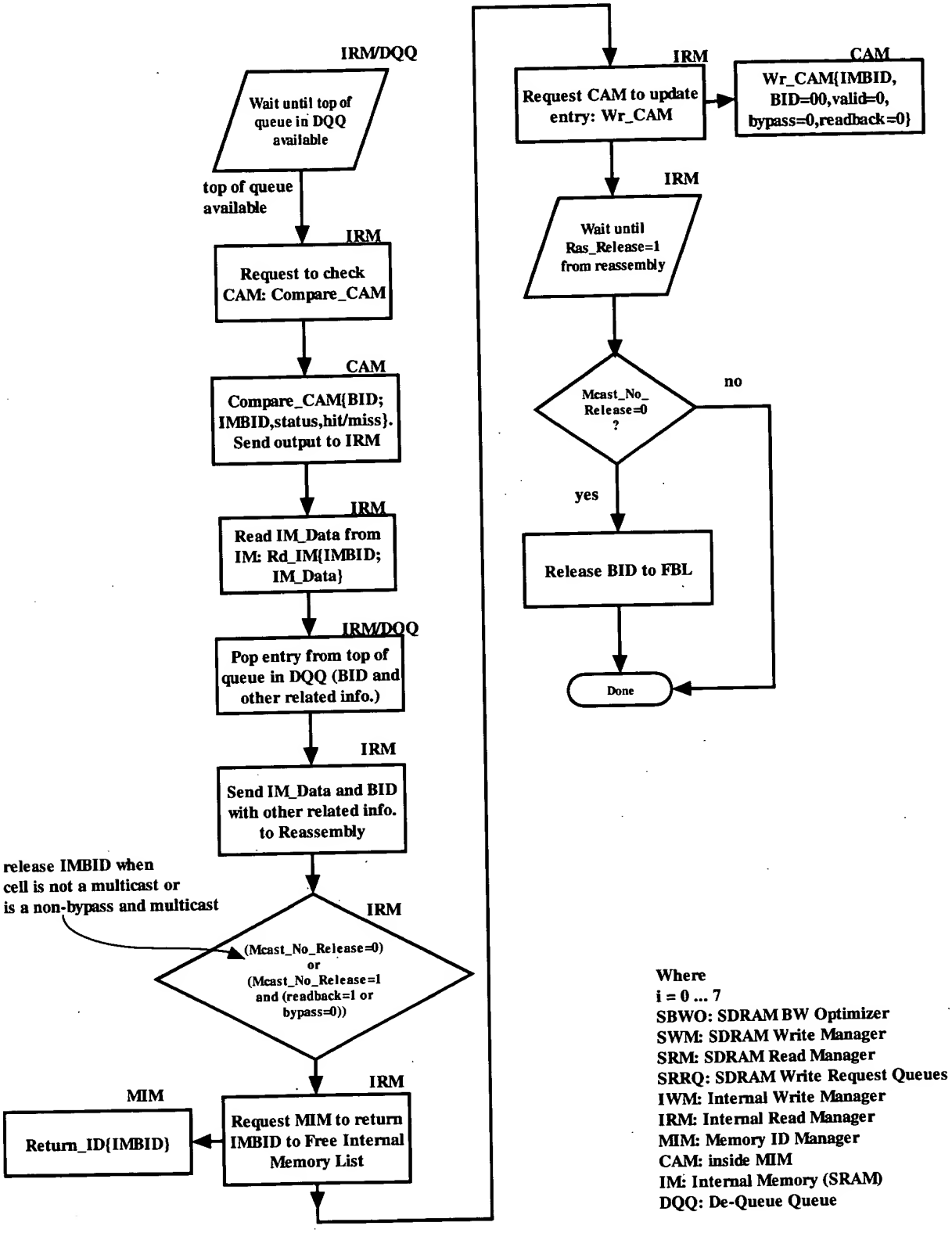


Figure 204

Interface with Reassembly



Where
i = 0 ... 7
SBWO: SDRAM BW Optimizer
SWM: SDRAM Write Manager
SRM: SDRAM Read Manager
SRRQ: SDRAM Write Request Queues
IWM: Internal Write Manager
IRM: Internal Read Manager
MIM: Memory ID Manager
CAM: inside MIM
IM: Internal Memory (SRAM)
DQQ: De-Queue Queue

Figure 205

RE-AS Type	Type	Application	Input	Memory (64B Cell)	Output	RE-ASSEMBLY/ TRANSLATE	CRC 32
1	0	Ingress TM (ATM=>ATM)	ATM	52B ATM +12B PAD	Switch	Off, Add Switch Header, Insert EFCI & CLP in Our Header	Off
1	1	IngressTM (ATM=>Packet)	AAL5	48B ATM +16B PAD	Switch	Off, Add Switch Header, Insert EFCI & CLP in Our Header	Off
1	2	IngressTM (Packet=>ATM)	Packet	48B ATM +16B PAD	Switch	Off, Add Switch Header	Off
1	3	Ingress TM (Packet=>Packet)	Packet	64B Cells (AAL5 Like)	Switch	Off, Add Switch Header	Off
2	4	ATM Encapsulation	ATM	52B ATM +12B PAD	Packet	Map to Packet, Add Switch Packet Header, Pass EFCI & CLP	Off
3	5	Reassembly, AAL-5 Trailer CRC Check.	AAL-5	48B ATM +16B PAD	Packet	Reassemble, Add Switch Packet Header, Pass EFCI & CLP, Check AAL5 CRC, Remove AAL5 Encap, Optional L2 Header Strip	On
4	6	Ingress Packet Bypass (Traffic Shaper)	Packet	64B Cells (AAL5 Like)	Packet	Reassemble, Add Switch Packet Header	Off
1	7	Messaging Cells	Serial	64B Cells	Sw/Pkt	Off, Add Switch Header	Off
5	8	Egress TM (ATM=> ATM)	Switch	52B ATM +12B PAD	ATM	Strip Pad, ATM Header Translate, Replace CLP & EFCI	Off
6	9	EgressTM (ATM=>Packet)	Switch	48B ATM +16B PAD	Packet	Reassemble, MPLS Manipulation, Decrement MPLS TTL, Check AAL5 CRC	On
7	10	EgressTM (Packet=>ATM)	Switch	48B ATM +16B PAD	AAL5	Strip Pad, Add ATM Header, Mark EOP using PTI, Replace CLP & EFCI	Off
8	11	Egress TM (Packet=>Packet)	Switch	64B Cells (AAL5	Packet	Reassemble, MPLS	On

				Like)		Manipulation, Decrement MPLS TTL, Check AAL5 CRC	
5	12	ATM De-Encapsulation	Packet	52B ATM +12B PAD	ATM	Strip Pad, ATM Header Translate, Replace CLP & EFCI	Off
7	13	Segmentation	Packet	48B ATM +16B PAD	AAL-5	Strip Pad, Add ATM Header, Mark EOP using PTI, Replace CLP & EFCI	Off
8	14	Egress Packet Bypass (Traffic Shaper)	Packet	64B Cells (AAL5 Like)	Packet	Reassemble, MPLS Manipulation, Decrement MPLS TTL, Check AAL5 CRC	On
	15	Unused					

Figure 206

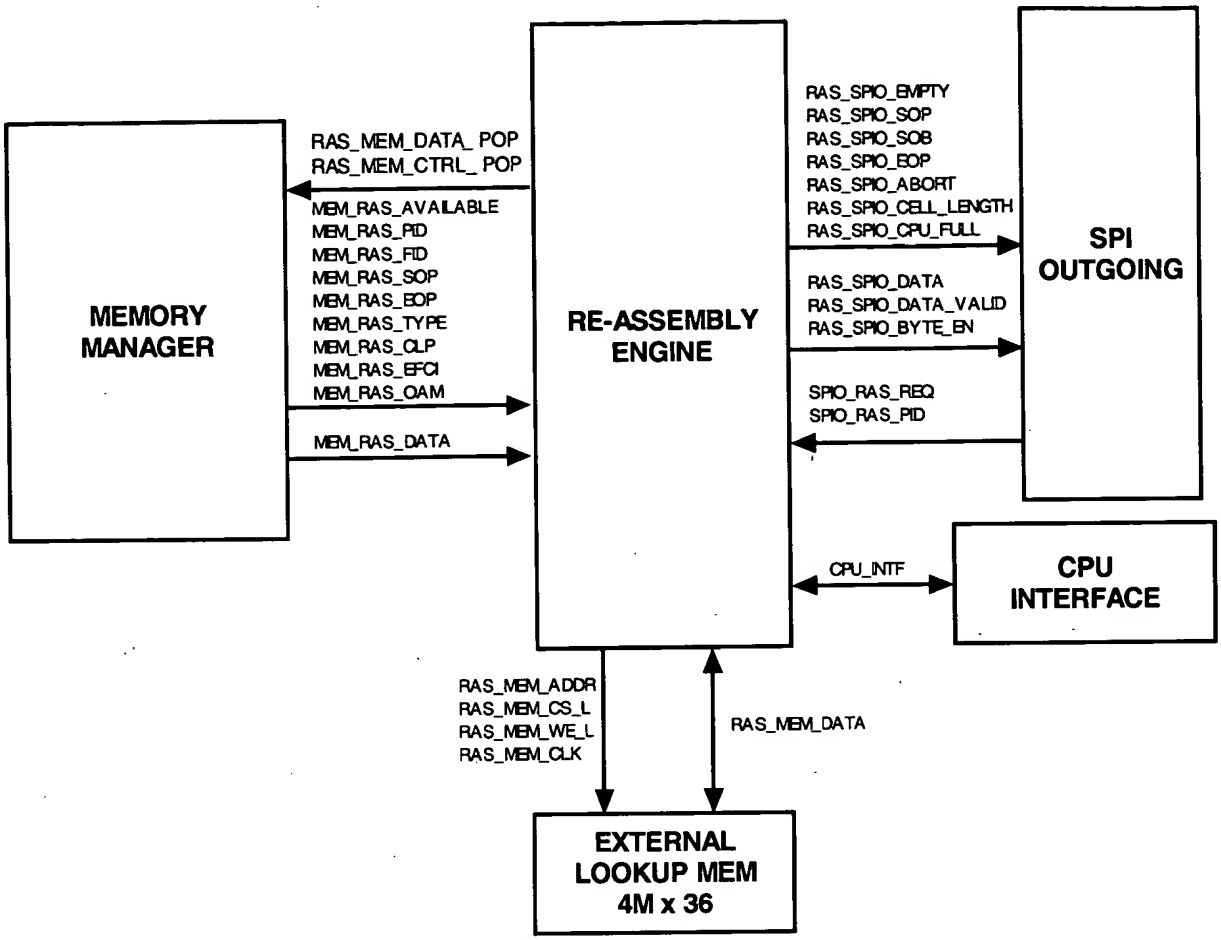


Figure 207

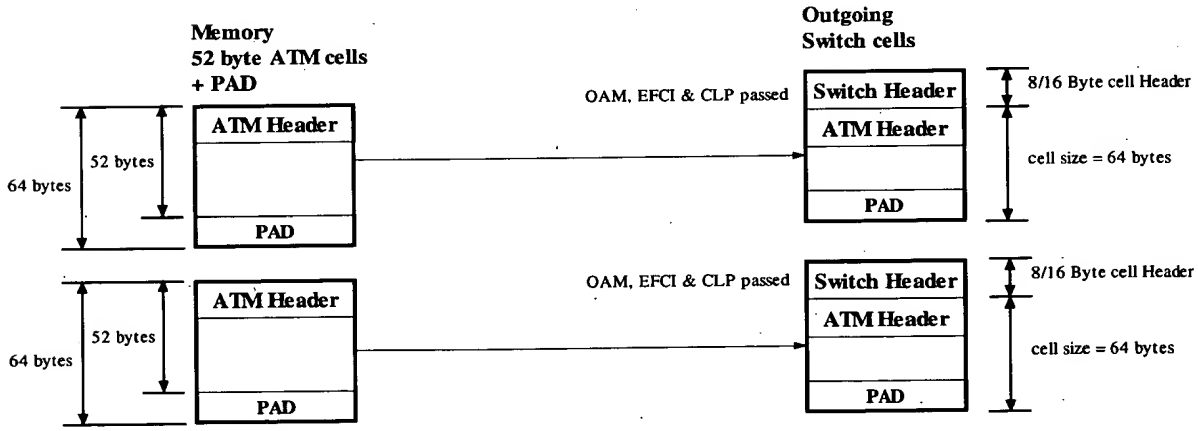


Figure 208

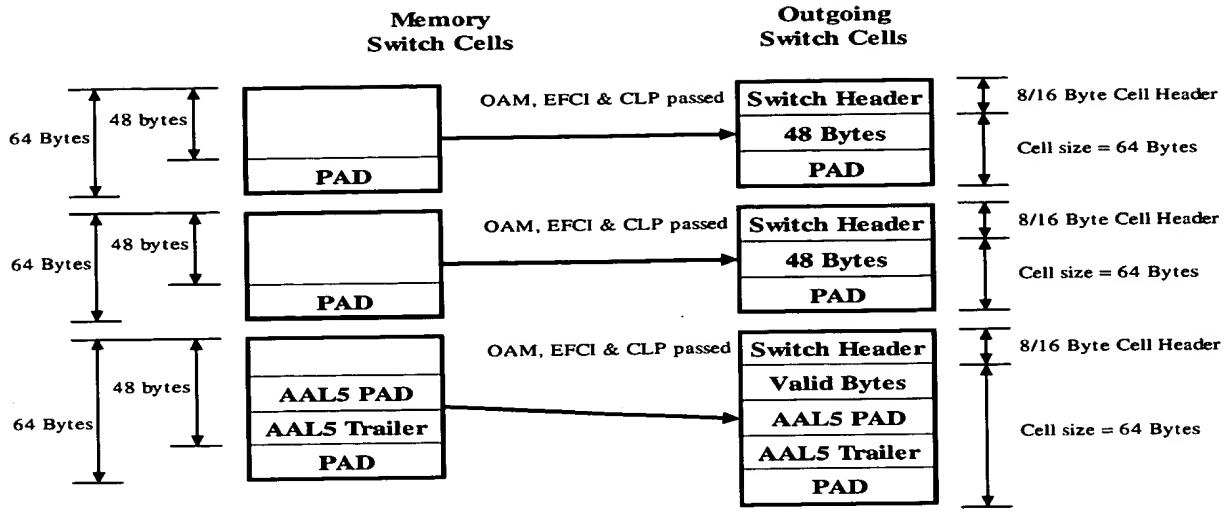


Figure 209

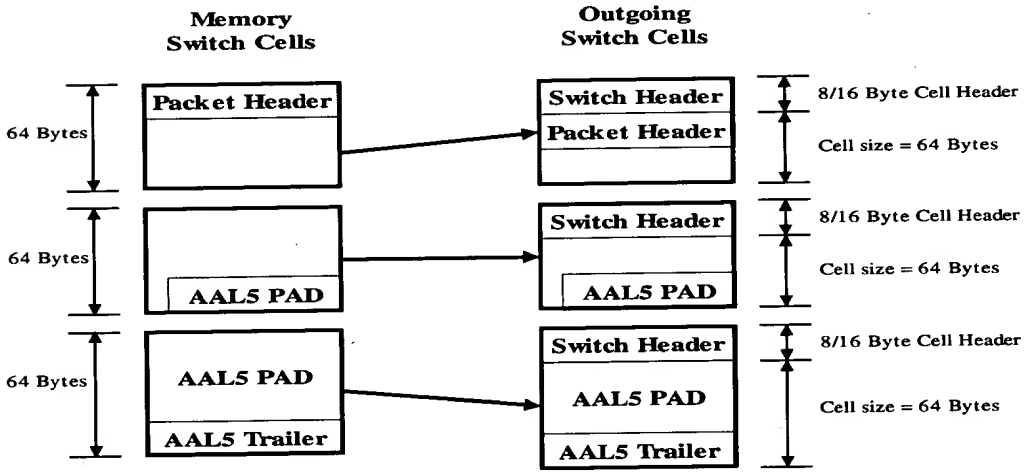


Figure 210

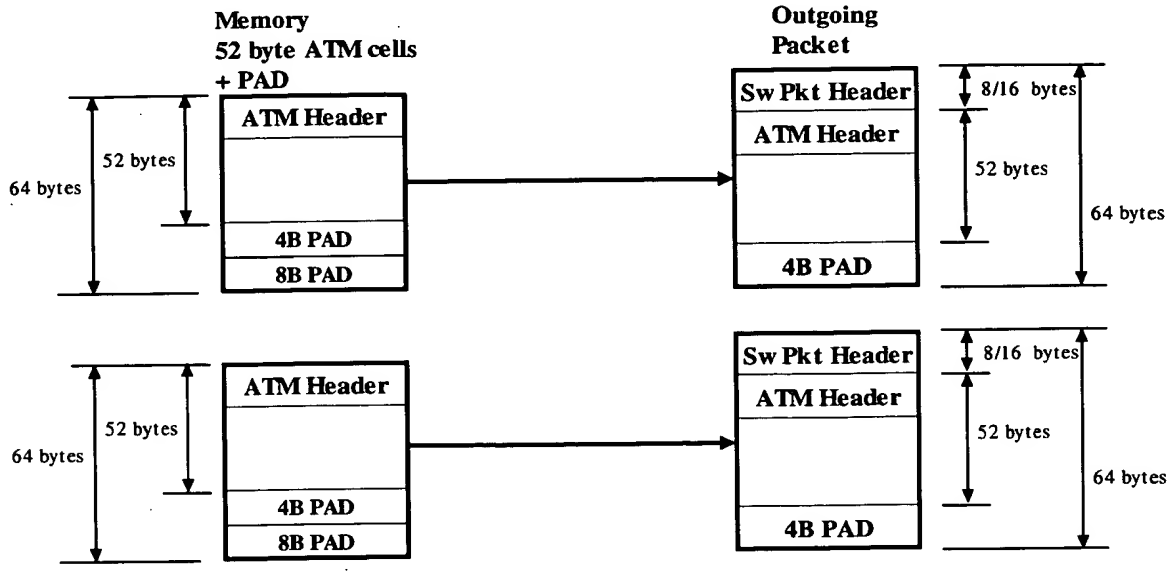


Figure 211

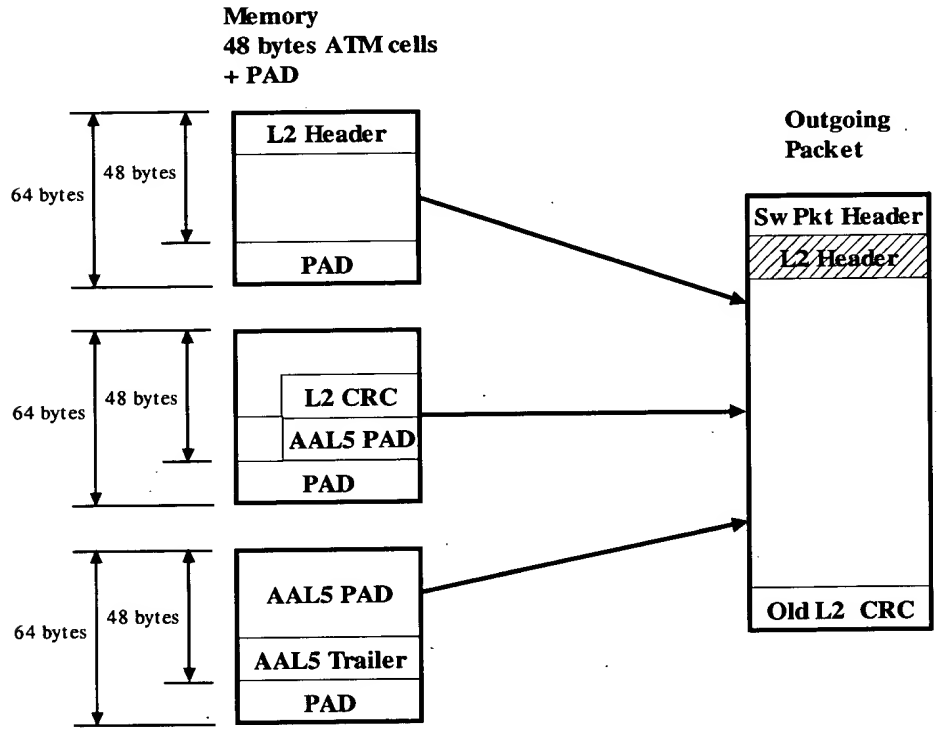


Figure 212

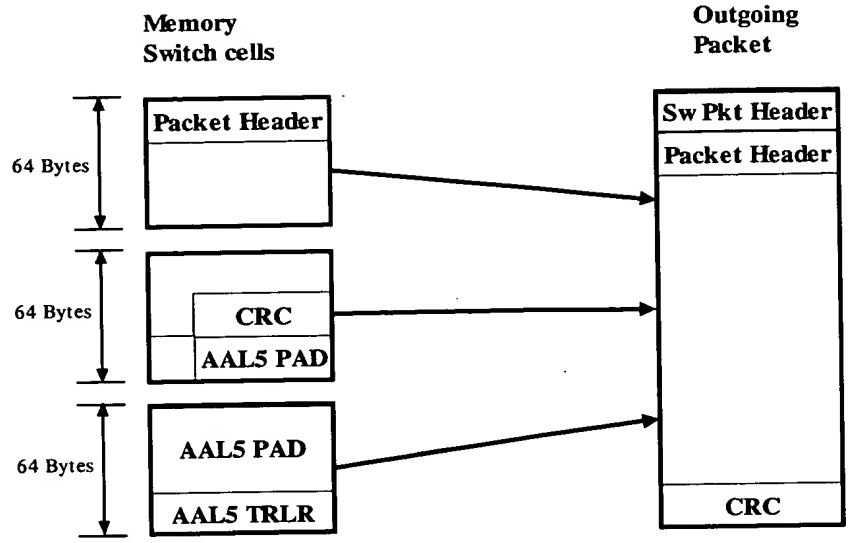


Figure 213

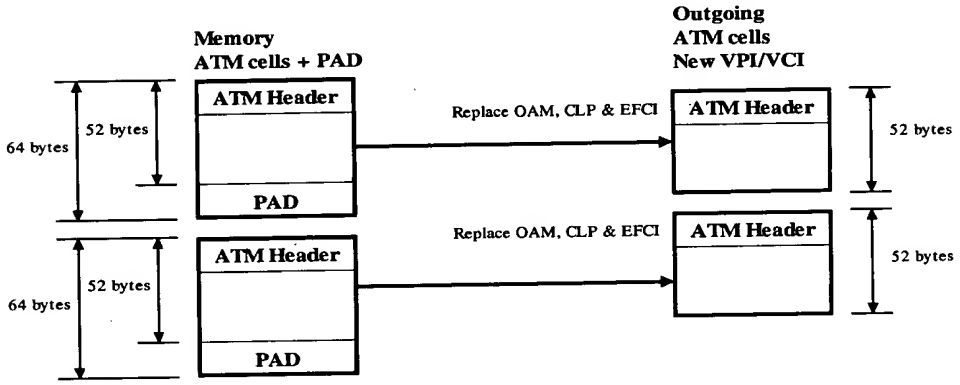


Figure 214

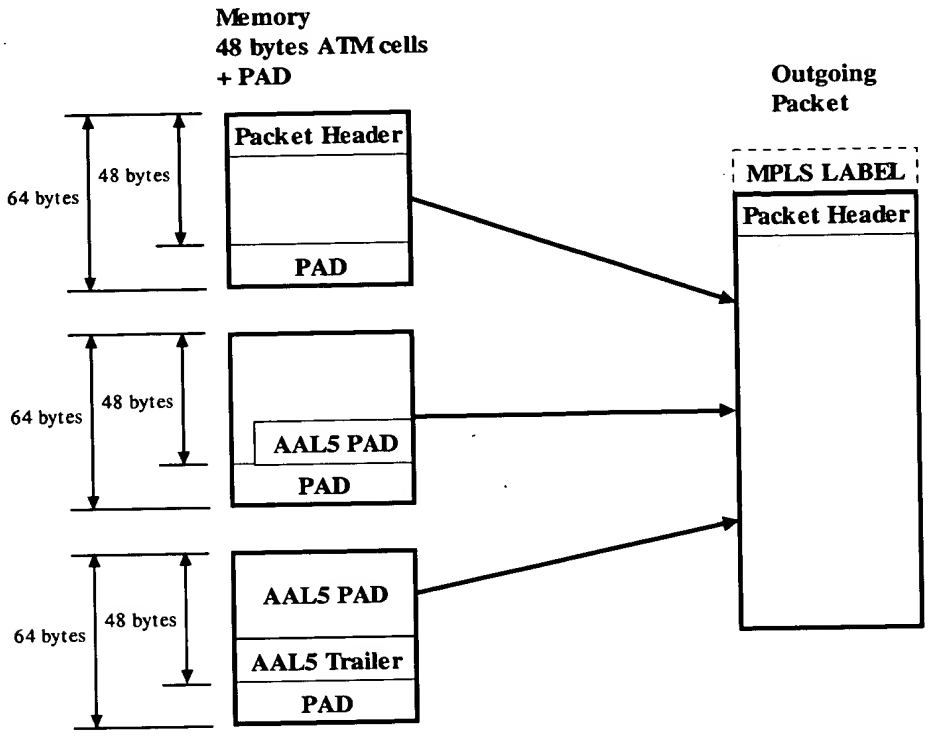


Figure 215

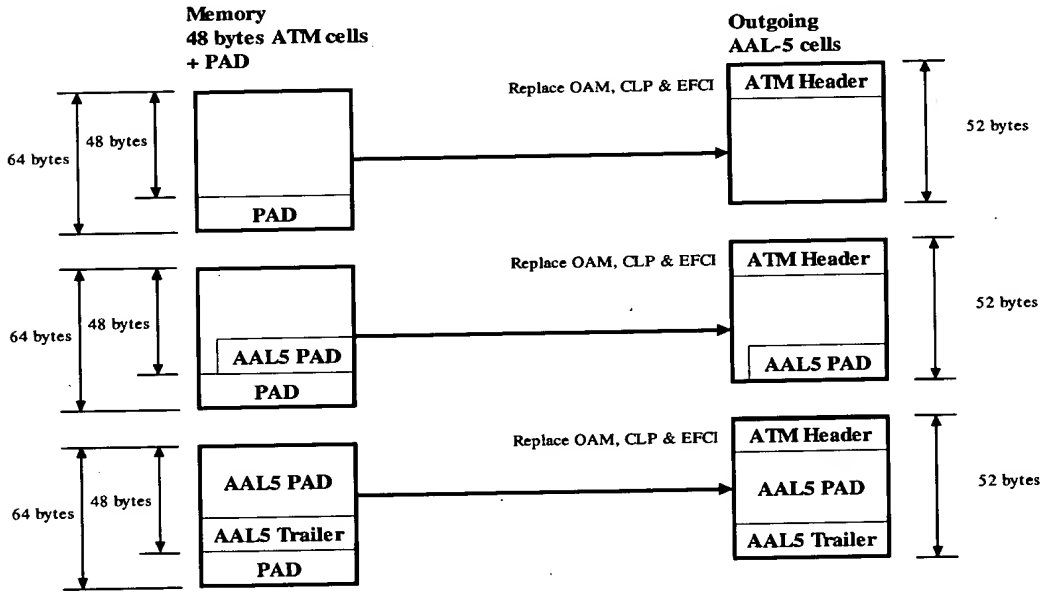


Figure 216

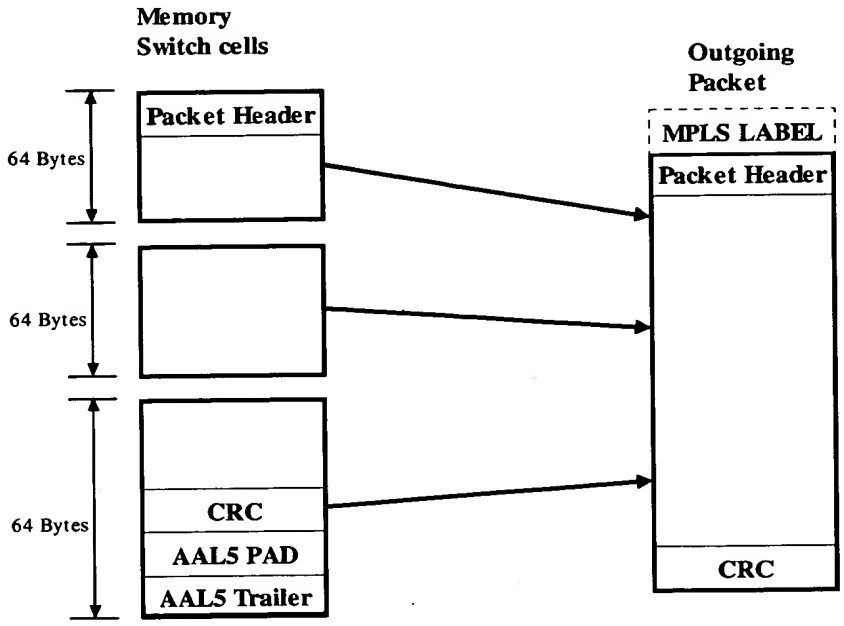


Figure 217

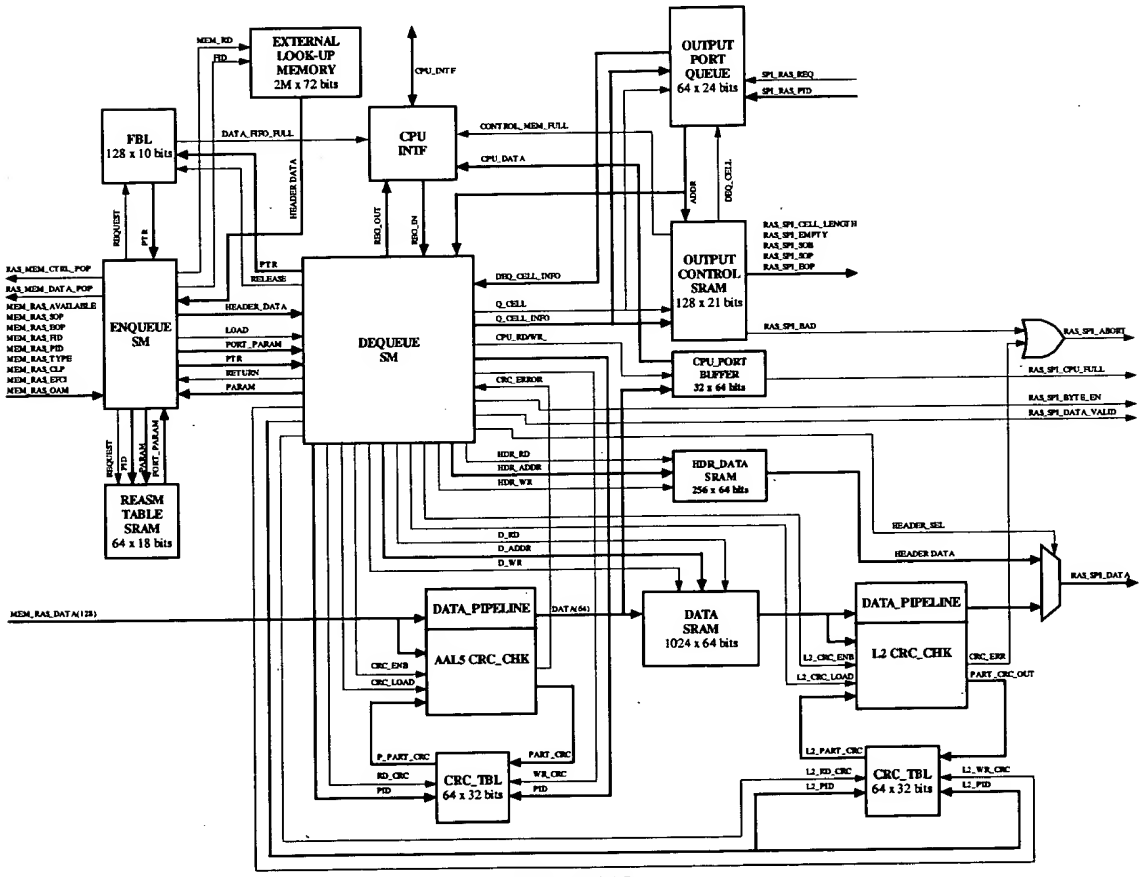


Figure 218

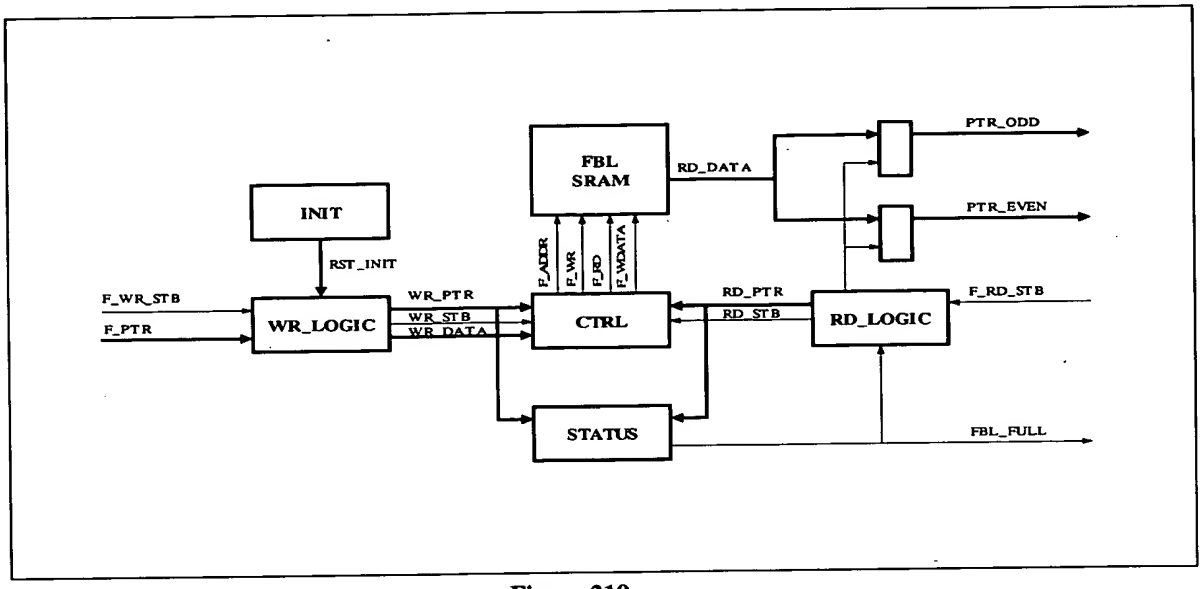


Figure 219

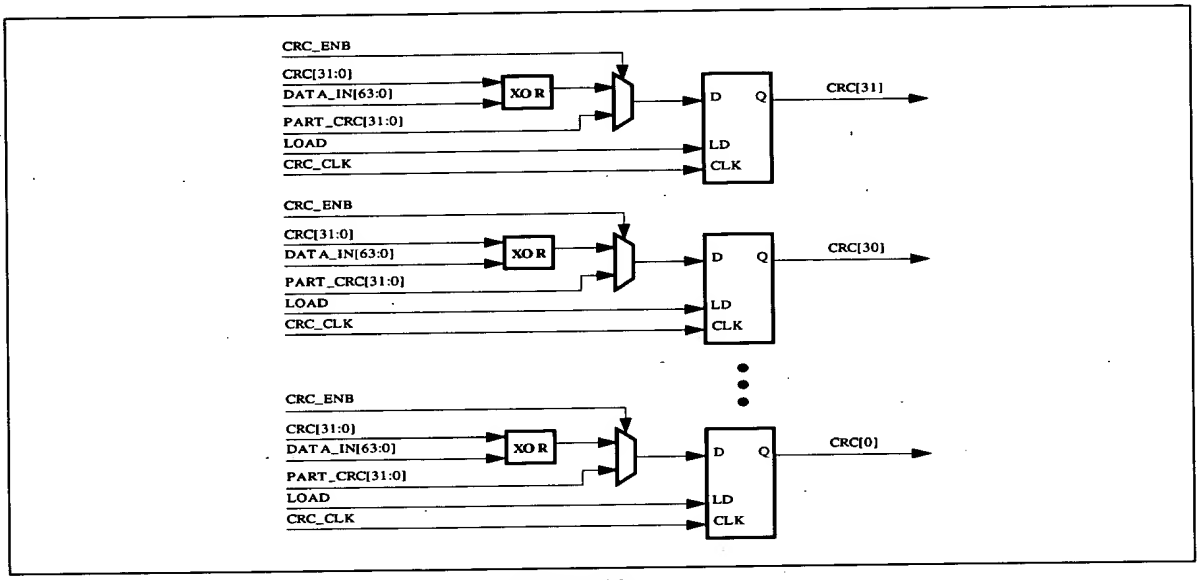


Figure 220

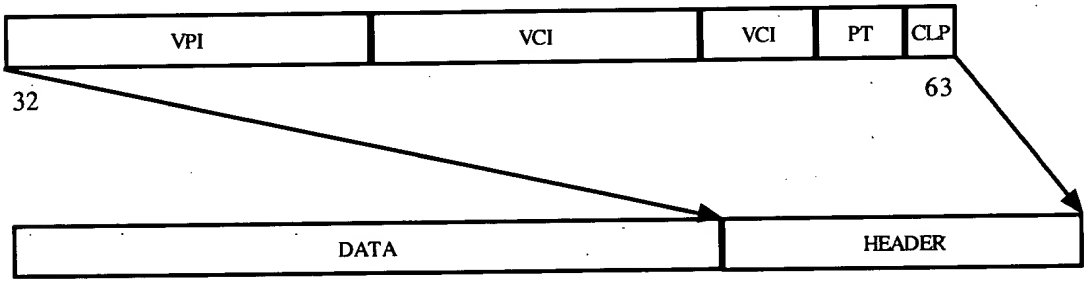


Figure 221

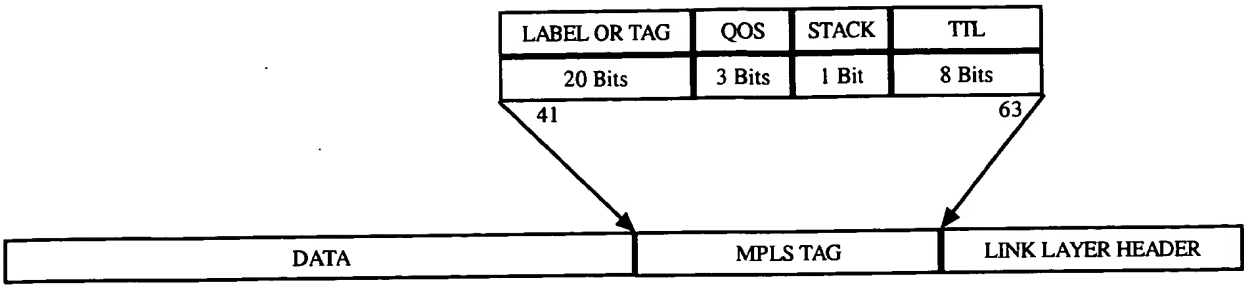


Figure 222

EFCI	1	7
CLP	1	6
OAM	1	5

Figure 223

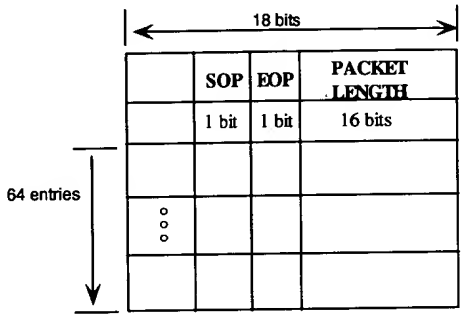


Figure 224

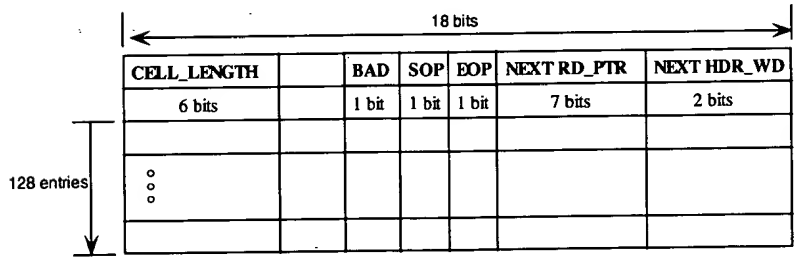


Figure 225

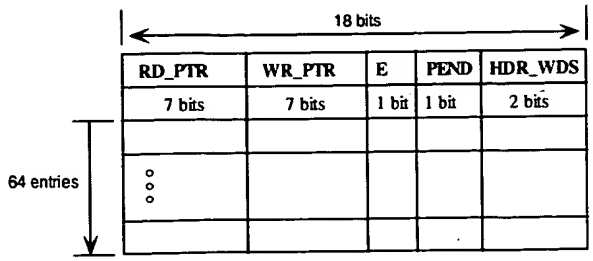


Figure 226

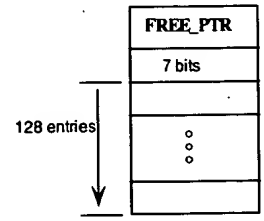


Figure 227

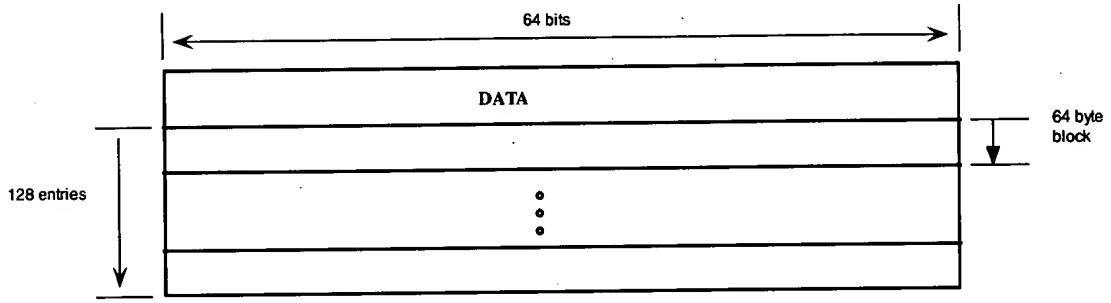


Figure 228

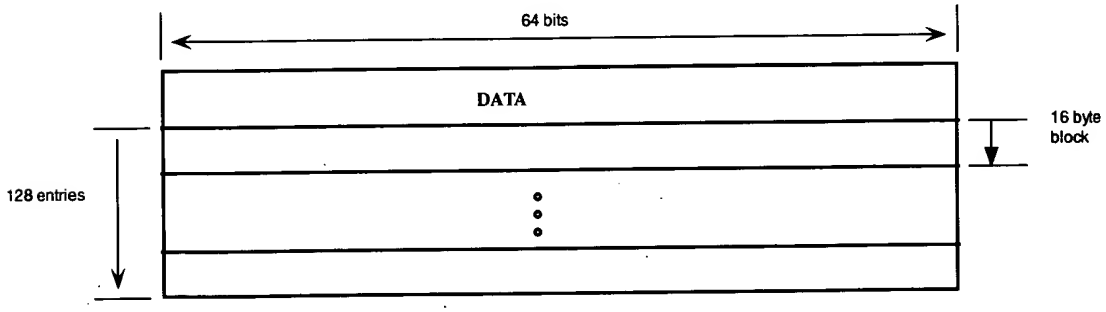


Figure 229

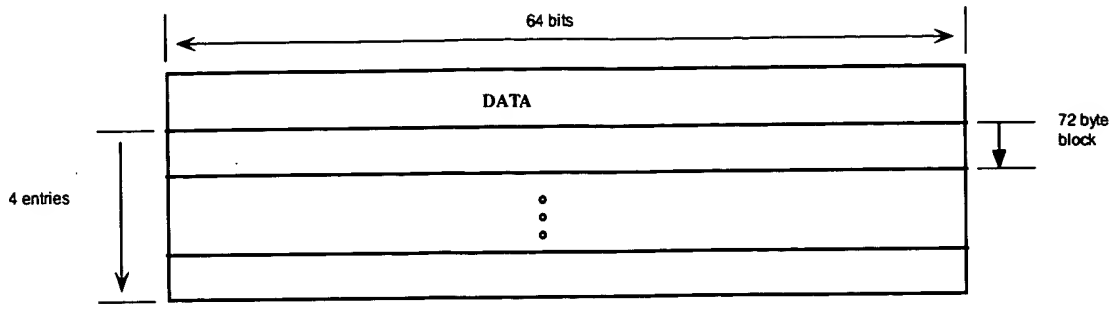


Figure 230

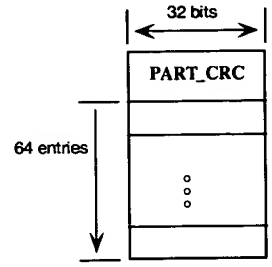


Figure 231

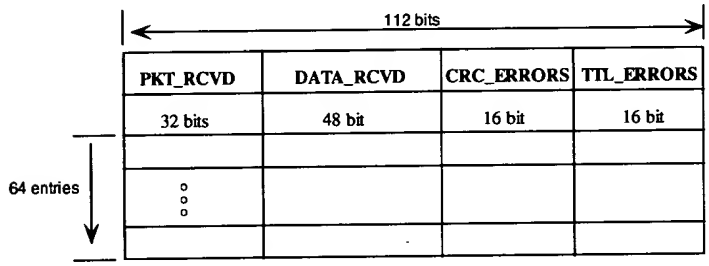


Figure 232

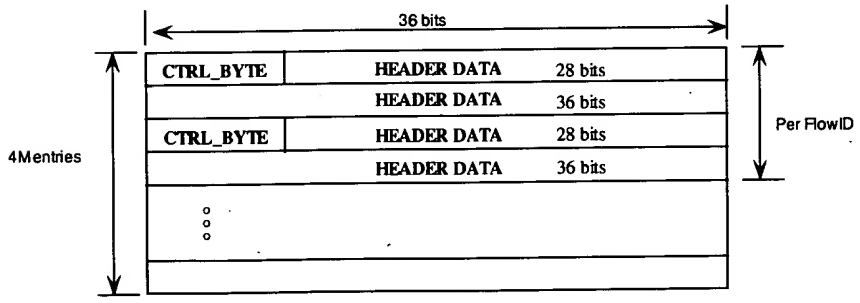


Figure 233

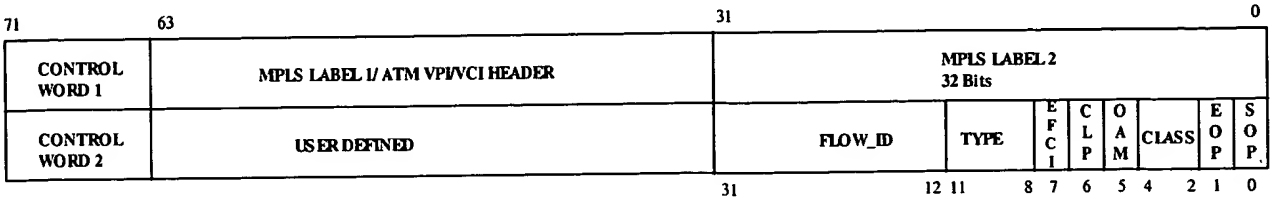


Figure 234

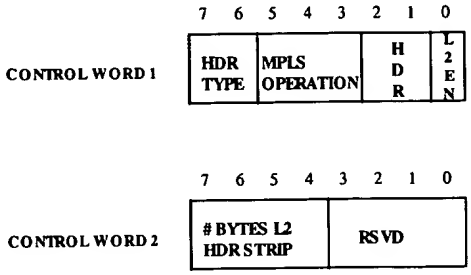


Figure 235

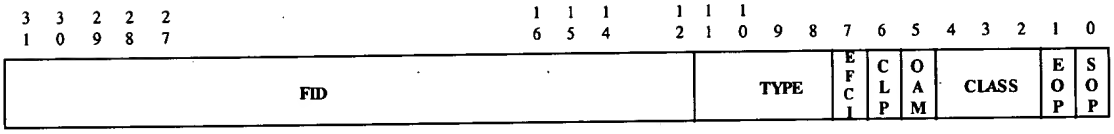


Figure 236

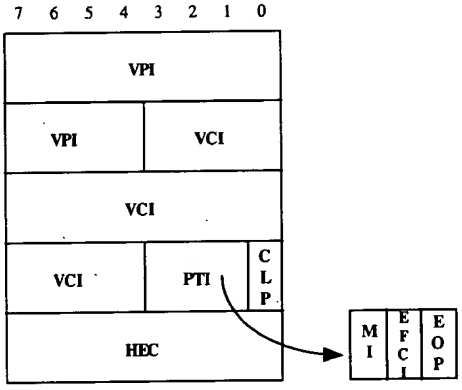


Figure 237

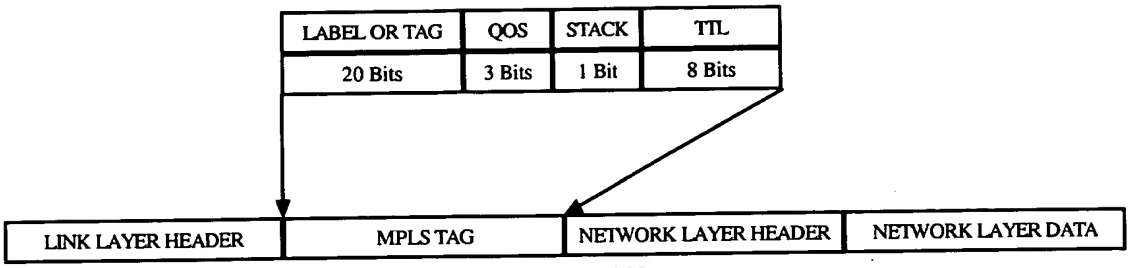


Figure 238

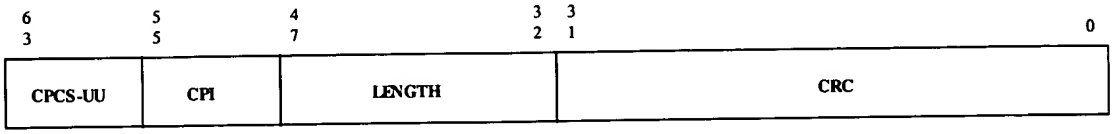


Figure 239

Signal Name	#bits	DIR	Description
RAS_CLK	1	IN	200 MHz internal system clock to RAS block.
RAS_RST_L	1	IN	A combination of POR and RAS Soft Reset.
G_SYNC	1	IN	Global Sync from the Global Sync block.
RAS_IN_EN	1	IN	Reassembly Input Enable signal.
RAS_OUT_EN	1	IN	Reassembly Output Enable Signal.
RAS_TST_MUX_OUT	32	OUT	Reassembly Test Mux pins.

Figure 240

Signal Name	#bits	DIR	Description
MEM_RAS_AVAILABLE	1	IN	Memory Manager has a cell available
MEM_RAS_SOP	1	IN	Start of Packet
MEM_RAS_EOP	1	IN	End of Packet
MEM_RAS_TYPE	4	IN	Type of traffic
MEM_RAS_PID	7	IN	Port number for output data (MSb marks CPU)
MEM_RAS_FID	20	IN	Flow ID
MEM_RAS_CLP	1	IN	Used to load the CLP bit in ATM Header
MEM_RAS_EFCI	1	IN	Used to load the EFCI bit in ATM Header
MEM_RAS_DATA	128	IN	Data in
RAS_MEM_DATA_POP	1	OUT	Data FIFO read signal from Reassembly to MM
RAS_MEM_CTRL_POP	1	OUT	Control FIFO read signal from Reassembly to MM

Figure 241

Signal Name	#bits	DIR	Description
SPIO_RAS_REQ	1	IN	SPI Requests a Data Cell
SPIO_RAS_PID	6	IN	SPI Port that is requesting data
RAS_SPIO_CPU_FULL	1	OUT	CPU DATA Buffer is full
RAS_SPIO_SOP	1	OUT	Start of Packet to SPI
RAS_SPIO_SOB	1	OUT	Start of Burst to SPI
RAS_SPIO_EOP	1	OUT	End of Packet to SPI
RAS_SPIO_ABORT	1	OUT	Indicate packet is to be discarded
RAS_SPIO_DATA	64	OUT	Data to SPI
RAS_SPIO_BYTE_EN	8	OUT	# of bytes of valid data to SPI
RAS_SPIO_DATA_VALID	1	OUT	Data is valid to SPI

Figure 242

Signal Name	#bits	DIR	Description
CPU_RAS_CS_L	1	IN	CPU RAS Block Select
CPU_RDWR_L	1	IN	CPU Read/Write_L
CPU_ADDR	6	IN	CPU Address
CPU_DATA_IN	32	IN	CPU Data In
RAS_CPU_DATA_OUT	32	OUT	CPU RAS Data Out

Figure 243

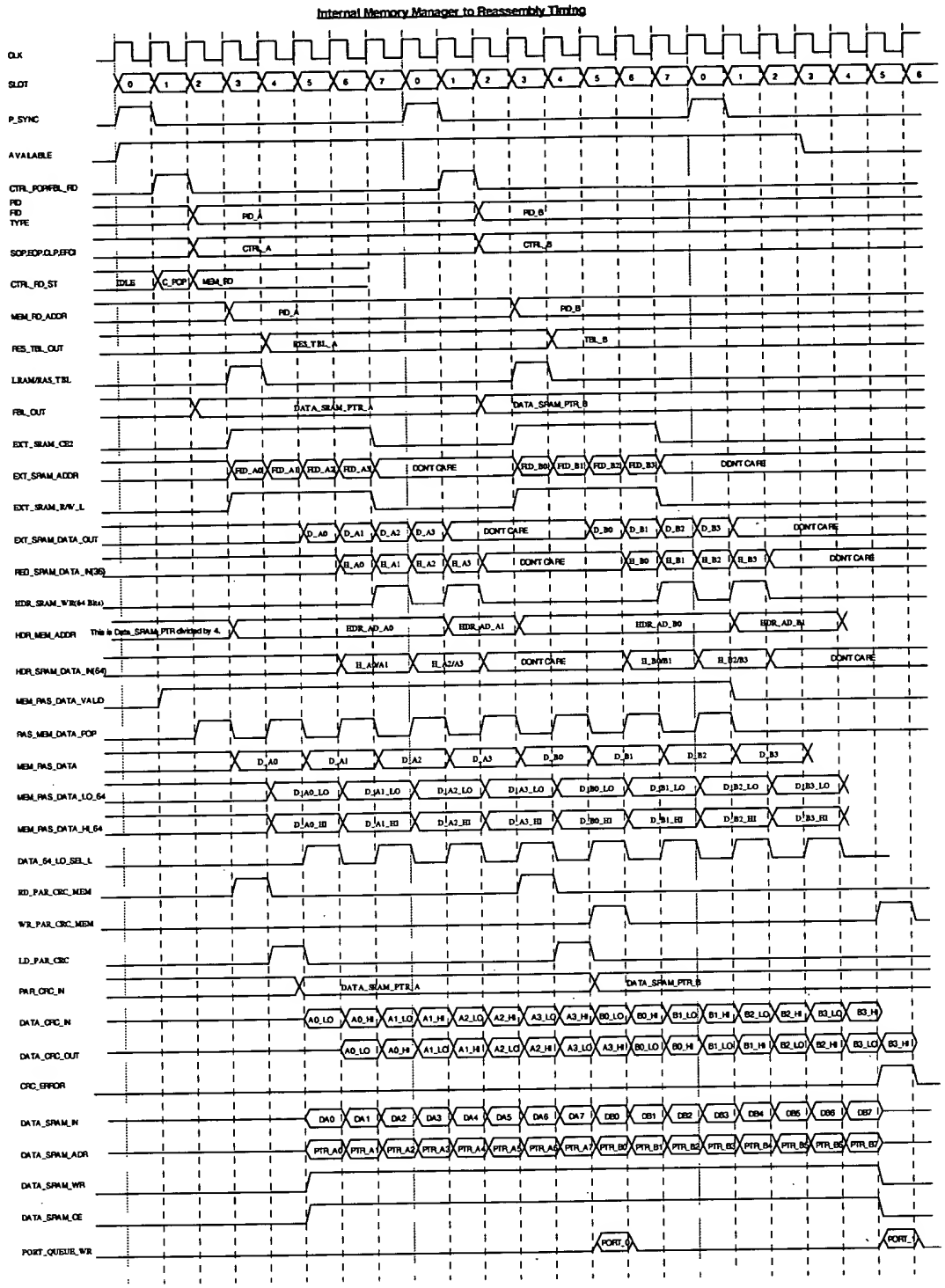

Internal Memory Manager to Reassembly Timing
CLK
SLOT
P_SYNC
AVAILABLE
CTRL_POP/FBL_RD
RD TYPE
SOP,EOP,CLP,EFCI
CTRL_RD_ST
MEM_RD_ADDR
RES_TBL_OUT
LRAM/BAS_TBL
FBL_OUT
EXT_SRAM_CE2
EXT_SRAM_ADDR
EXT_SRAM_R/W_L
EXT_SRAM_DATA_OUT
RED_SRAM_DATA_IN(36)
HDR_SRAM_WR(64 Bits)
HDR_MEM_ADDR
HDR_SRAM_DATA_IN(64)
MEM_RAS_DATA_VALID
RAS_MEM_DATA_POP
MEM_RAS_DATA
MEM_RAS_DATA_LO_64
MEM_RAS_DATA_HI_64
DATA_64_LO_SEL_L
RD_PAR_CRC_MEM
WR_PAR_CRC_MEM
LD_PAR_CRC
PAR_CRC_IN
DATA_CRC_IN
DATA_CRC_OUT
CRC_ERROR
DATA_SRAM_IN
DATA_SRAM_ADR
DATA_SRAM_WR
DATA_SRAM_OE
PORT_QUEUE_WR
IDLE
C_POP
MEM_RD
RD_A
RD_B
CTRL_A
CTRL_B
DATA_SRAM_PTR_A
DATA_SRAM_PTR_B
DON'T CARE
EDR_AD_A0
EDR_AD_A1
EDR_AD_B0
EDR_AD_B1
PORT_0
PORT_1

Figure 244

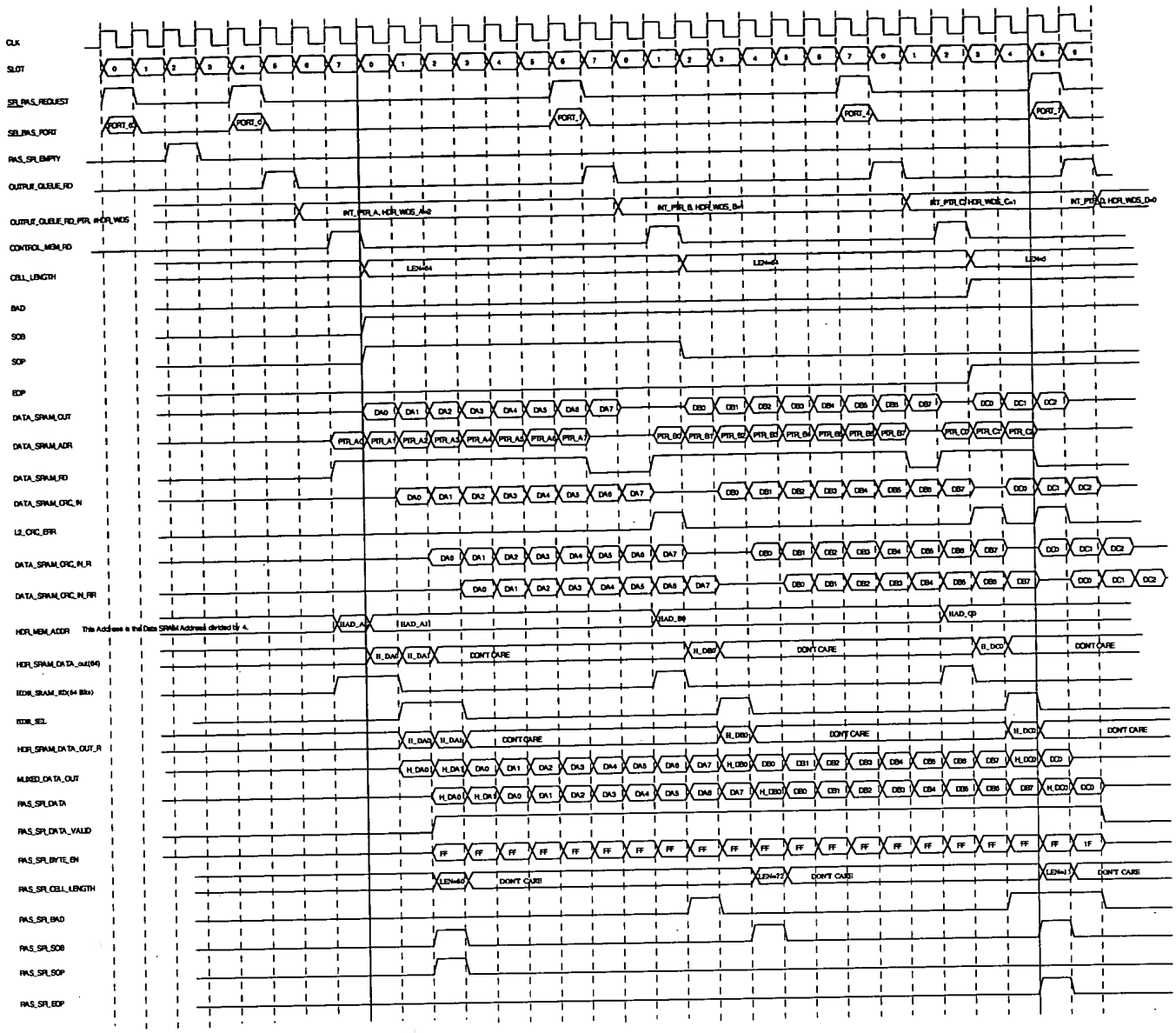

CLK
SLOT
SRI_PAS_REQUEST
SRI_PAS_PORT
PAS_SRI_EMPTY
OUTPUT_QUEUE_RD
OUTPUT_QUEUE_RD_PTR_&HDR_WDS
CONTROL_MEM_RD
CELL_LENGTH
BAD
SOB
SOP
EOP
DATA_SRAM_OUT
DATA_SRAM_ADR
DATA_SRAM_RD
DATA_SRAM_CRC_IN
L2_CRC_ERR
DATA_SRAM_CRC_N_R
DATA_SRAM_CRC_N_RR
HDR_MEM_ADDR
This Address is the Data SRAM Address divided by 4.
HDR_SRAM_DATA_out(64)
HDR_SRAM_RD(64 Bits)
HDR_SEL
HDR_SRAM_DATA_OUT_R
MUXED_DATA_OUT
PAS_SRI_DATA
PAS_SRI_DATA_VALID
PAS_SRI_BYTE_EN
PAS_SRI_CELL_LENGTH
PAS_SRI_BAD
PAS_SRI_SOB
PAS_SRI_SOP
PAS_SRI_EOP
DON'T CARE

Figure 245

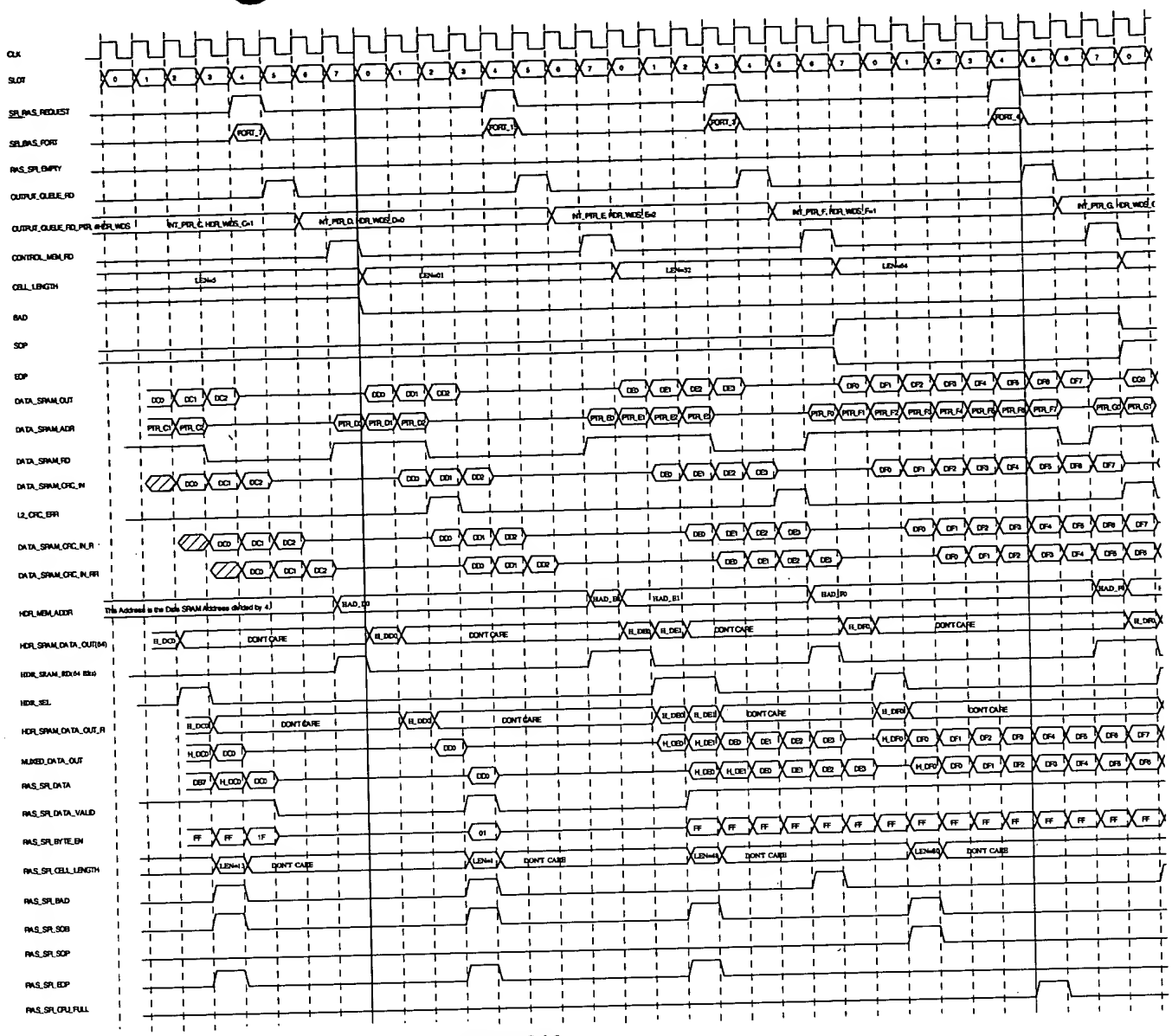

CLK
SLOT
SR_RAS_REQUEST
SR_RAS_PORT
RAS_SR_EMPTY
OUTPUT_QUEUE_RD
OUTPUT_QUEUE_RD_PTR_WDS
CONTROL_MEM_RD
CELL_LENGTH
BAD
SOP
EOP
DATA_SRAM_OUT
DATA_SRAM_ADR
DATA_SRAM_RD
DATA_SRAM_CRC_IN
L2_CRC_ERR
DATA_SRAM_CRC_IN_R
DATA_SRAM_CRC_IN_RR
HDR_MEM_ADDR
HDR_SRAM_DATA_OUT(64)
HDR_SRAM_RD(64 BIT)
HDR_SEL
HDR_SRAM_DATA_OUT_R
MUXED_DATA_OUT
RAS_SR_DATA
RAS_SR_DATA_VALID
RAS_SR_BYTE_EN
RAS_SR_CELL_LENGTH
RAS_SR_BAD
RAS_SR_SOB
RAS_SR_SOP
RAS_SR_EOP
RAS_SR_CPU_FULL
This Address is the Data SRAM Address divided by 4
DON'T CARE

Figure 246

Input phase to RAS from Internal Memory Manager

Memory	0	1	2	3	4	5	6	7	0	1	2	3	4	5
MM Ctrl Mem		R1								R2				
MM Data Mem			R1		R1		R1		R1		R2		R2	
HDR Memory				R1	R1	R1	R1					R2	R2	R2
FBL Memory		R1								R2				
RAS Table				R1		W1					R2			W2
Data Memory						W1	W1	W1	W1	W1	W1	W1	W1	
Header Mem								W1		W1				
Par_CRC_Mem				R1								**R2**		W1
Out Cntrl Mem						W1				W0				
Out Port Queue								R1						W1
Statistics Mem								R1						W1
Cpu Data Mem						W1	W1	W1	W1	W1	W1	W1	W1	

Note: R2 for the Partial CRC Table is dependent on the PID. If the next cell has the same PID as the previous cell then this read information is not used..

Figure 247

Errors	Decode	Description
2 SOP, No EOP, same port		This error happens when 2 SOPs are received with no EOP between them for the same port.
No SOP, 2 EOP, same port		This error happens when 2 EOPs are received with no SOP between them for the same port.
Exceeds MTU		This error happens when the packet length is larger than the programmable maximum transfer unit.
CRC error		This happens when the generated CRC do not match those in the trailer for AAL-5 traffic in the others category.
PKT_LEN error		This happens when the calculated packet length does not match the one in the trailer.
Queue FIFO full		The Q_FIFO is full, no room to queue complete cells
Free pointer empty		The data FIFO is full, no room for incoming cells

Figure 248

Address	Name	Type	Description
0	COM	R/W	[31:28] – Opcode [27:0] – Address, depending on the command. No default value.
1	R0	R/W	[31:0] General-purpose register. No default value

2	R1	R/W	[63:32] General-purpose register. No default value
3	R2	R/W	[95:64] General-purpose register. No default value
4	R3	R/W	[127:96] General-purpose register. No default value
5	R4	R/W	[159:128] General-purpose register. No default value
6	R5	R/W	[191:160] General-purpose register. No default value
7	R6	R/W	[223:192] General-purpose register. No default value
8	R7	R/W	[255:224] General- purpose register. No default value
9	R8	R/W	[287:256] General-purpose register. No default value
10	R9	R/W	[319:288] General-purpose register. No default value
11	R10	R/W	[351:320] General-purpose register. No default value
12	R11	R/W	[383:352] General-purpose register. No default value
13	R12	R/W	[415:384] General-purpose register. No default value
14	R13	R/W	[447:416] General-purpose register. No default value
15	R14	R/W	[479:448] General-purpose register. No default value
16	R15	R/W	[511:480] General-purpose register. No default value
17 – 31	Reserved		
32	CONTROL	R/W	[1:0] – EXT_MEM_SIZE 00 = 256K 01 = 512K (Default) 10 = 1M 11 = Reserved [31:2] – Reserved

33	STATUS/MASK	R/W	[0] – TWO_SOP_ERR Error when two SOPs occur with no EOP between for a port. Default value 0 (No Error). [1] – TWO_EOP_ERR Error when two EOPs occur with no SOP between for a port. Default value 0 (No Error). [2] – EXD_MTU Error when packet exceeds 16KB. Default value 0 (No Error). [3] – CRC_ERR Error for AAL5 CRC mismatch. Default value 0 (No Error). [4] – LEN_ERR Error for AAL5 Length mismatch. Default value 0 (No Error). [5] – FUL_ERR Error for Output Queue Full. Default value 0 (No Error). [6] – CPU_DATA CPU has a Data Cell available. Default value 0 (No Cell). [7] – Reserved [8] – TWO_SOP_ERR_MASK Mask when two SOPs occur with no EOP between for a port Error. Default value 0 (No Mask). [9] – TWO_EOP_ERR_MASK Mask when two EOPs occur with no SOP between for a port Error. Default value 0 (No Mask). [10] – EXD_MTU_MASK Mask packet exceeds 16KB Error. Default value 0 (No Mask). [11] – CRC_ERR_MASK Mask for AAL5 CRC Error. Default value 0 (No Mask). [12] – LEN_ERR_MASK Mask for AAL5 Length Error. Default value 0 (No Mask). [13] – FUL_ERR_MASK Mask Output Queue FIFO Full Error. Default value 0 (No Mask). [31:14] – Reserved.

34	TEST MUX	R/W	[3:0] – TEST GROUP SELECT 0000 = No Output (Default) 0001 = Group 1 0010 = Group 2 0011 = Group 3 Others = Reserved: No output [31:4] – Reserved.
35 - 63	Reserved		

Figure 249

Opcode	Command Name	Description
0	NOP	
1	Rd RAS ext mem	36 bits, data to registers R0, R1, R2, R3, R4, R5 (Up to 4M locations)
2	Wr RAS ext mem	Take data from registers R0, R1, R2, R3, R4, R5
3	Rd Data mem	64 bits, data to registers R0, R1 (Up to 1024 locations)
4	Wr Data mem	Take data from registers R0, R1
5	Rd crc1, crc2, ras, outq mem	Read 4 memories in one command. 106 bits, data to registers R0, R1, R2, R3 (Up to 64 locations)
6	Wr crc1, crc2, ras, outq mem	Write to 4 memories in one command. Take data from registers R0, R1, R2, R3
7	Rd stat mem	96 bits, data to registers R0, R1, R2 (Up to 64 locations)
8	Wr stat mem	Take data from registers R0, R1, R2
9	Rd CPU mem	64 bits, data to registers R0, R1 (Up to 32 locations)
10	Wr CPU mem	Take data from registers R0, R1
11	Rd FBL, Out Ctrl mem	Read 2 memories in one command. 31 bits, data to register R0 (Up to 128 locations)
12	Wr FBL, Out Ctrl mem	Write to 2 memories in one command. Take data from register R0
13	Rd HDR mem	64 bits, data to registers R0, R1 (Up to 256 locations)
14	Wr HDR mem	Take data from registers R0, R1
15	Get CPU cell	Opcode only, no address, the cell is transferred to R0-R15. This command causes a POP from that FIFO while the regular Rd/Wr commands just access specific locations.
16	Init Memories	Opcode only, no address, The command causes the Reassembly block to initialize the Free Buffer List and the Output Port Queue.

Figure 250

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00001	R[6:0]		FID_ADDR[19:0]				
R0	R[23:0]						CTRL_WD1[7:0]	
R1	WD1_HDR_DATA[63:32]							
R2	WD1_HDR_DATA[31:0]							
R3	R[23:0]						CTRL_WD2[7:0]	
R4	WD2_HDR_DATA[63:32]							
R5	WD2_HDR_DATA[31:0]							

Figure 251

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00010	R[6:0]		FID_ADDR[19:0]				
R0	R[23:0]						CTRL_WD1[7:0]	
R1	WD1_HDR_DATA[63:32]							
R2	WD1_HDR_DATA[31:0]							
R3	R[23:0]						CTRL_WD2[7:0]	
R4	WD2_HDR_DATA[63:32]							
R5	WD2_HDR_DATA[31:0]							

CTRL_WD1[0] = RSVD
CTRL_WD1[1] = TTLD
CTRL_WD1[2] = HDR BYTES
CTRL_WD1[5:3] = MPLS_OP[2:0]
CTRL_WD1[7:6] = HDR_TYPE[1:0]

CTRL_WD2[3:0] = RSVD[3:0]
CTRL_WD2[7:4] = HDR_BYTES_STRIP[3:0]

Figure 252

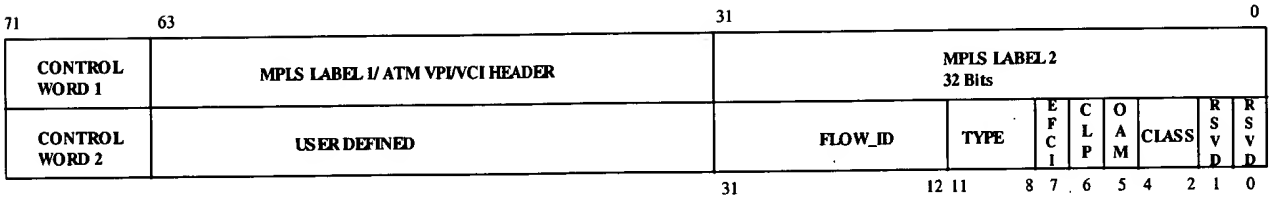


Figure 253

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00011	R[16:0]					Addr[9:0]	
R0	DATA[31:0]							
R1	DATA[63:32]							

Figure 254

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00100	R[16:0]					Addr[9:0]	
R0	DATA[31:0]							
R1	DATA[63:32]							

Figure 255

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00101	R[20:0]					Addr[5:0]	
R1	PARTIAL_CRC1[31:0]							
R1	PARTIAL_CRC2[31:0]							
R2	R[13:0]			PKT_LENGTH[17:2]			E	S
R3	R[7:0]			OUT_Q[23:0]				

Figure 256

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00110	R[20:0]					Addr[5:0]	
R1	PARTIAL_CRC1[31:0]							
R1	PARTIAL_CRC2[31:0]							
R2	R[13:0]			PKT_LENGTH[17:2]			E	S
R3	R[7:0]			OUT_Q[23:0]				

E – EOP
S – SOP

OUT_Q[0] = EMPTY
OUT_Q[1] = PENDING
OUT_Q[3:2] = HEADER_WORDS[1:0]
OUT_Q[13:4] = READ_POINTER[9:0]
OUT_Q[23:14] = WRITE_POINTER[9:0]

Figure 257

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	00111	R[20:0]						Addr[5:0]
R1	DATA_CNT[31:0]							
R2	PACKET_CNT[15:0]				DATA_CNT[48:32]			
R3	ERROR_CNT[15:0]				PACKET_CNT[31:16]			

Figure 258

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01000	R[20:0]						Addr[5:0]
R1	DATA_CNT[31:0]							
R2	PACKET_CNT[15:0]				DATA_CNT[48:32]			
R3	ERROR_CNT[15:0]				PACKET_CNT[31:16]			

Figure 259

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01001	R[21:0]						Addr[4:0]
R0	DATA[31:0]							
R1	DATA[63:32]							

Figure 260

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01010	R[21:0]						Addr[4:0]
R0	DATA[31:0]							
R1	DATA[63:32]							

Figure 261

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01011	R[19:0]					Addr[6:0]	
R0	R	OUT_CTRL[20:0]				FBL_PTR[9:0]		

Figure 262

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01100	R[19:0]					Addr[6:0]	
R0	R	OUT_CTRL[20:0]				FBL_PTR[9:0]		

R – Reserved

OUT_CTRL[5:0] = CELL_LENGTH[5:0]
OUT_CTRL [6] = BAD
OUT_CTRL [7] = SOP
OUT_CTRL [8] = EOP
OUT_CTRL [19:10] = NEXT RD_POINTER[9:0]
OUT_CTRL [21:20] = NEXT HDR_WORDS[1:0]

Figure 263

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01101	R[18:0]					Addr[7:0]	
R0	DATA[31:0]							
R1	DATA[63:32]							

Figure 264

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01110	R[18:0]					Addr[7:0]	
R0	DATA[31:0]							
R1	DATA[63:32]							

Figure 265

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	01111	R[27:0]						
R0	DATA[31:0]							
R1	DATA[63:32]							
R2	DATA[95:64]							
R3	DATA[127:96]							
R4	DATA[159:128]							
R5	DATA[191:160]							
R6	DATA[223:192]							
R7	DATA[255:224]							
R8	DATA[287:256]							
R9	DATA[319:288]							
R10	DATA[351:320]							
R11	DATA[383:352]							
R12	DATA[415:384]							
R13	DATA[447:416]							
R14	DATA[479:448]							
R15	DATA[511:480]							

Figure 266

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	10000	R[27:0]						

Figure 267

No.	Traffic Type	# of Tag bits	Tag word	Tag bits
1	ATM	28	1st 64-bit word	1-28
2 (a)	MPLS – ATM	20	1st 64-bit word	41-60
2 (b)	MPLS – PPP	20	1st 64-bit word	33-52 or 41-60 (depends on PPP mode)
2 (c)	MPLS – Ethernet	20	3rd & 4th 64-bit word	49-64 (3rd wd) & 1-4 (4th wd)
2 (d)	MPLS – FR	20	1st 64-bit word	25-44
3	Ethernet	48	2nd 64-bit word	1-48
4	IP	32	3rd 64-bit word	1-32
5	Frame Relay (FR)	10	1st 64-bit word	9-14 & 17-20

Figure 268

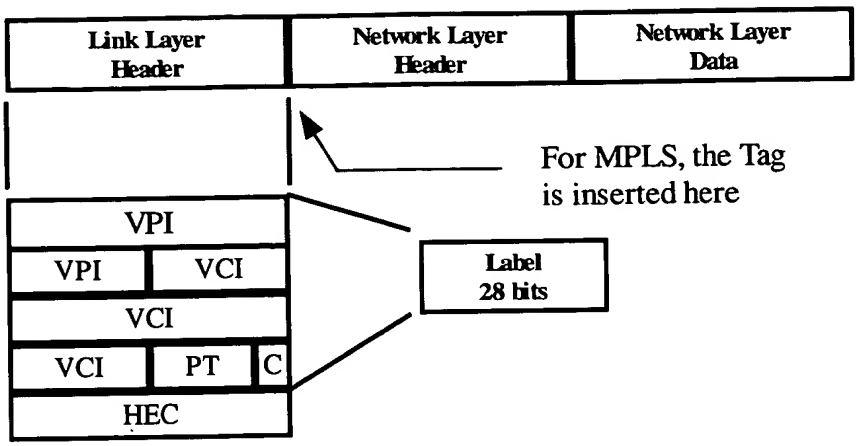


Figure 269

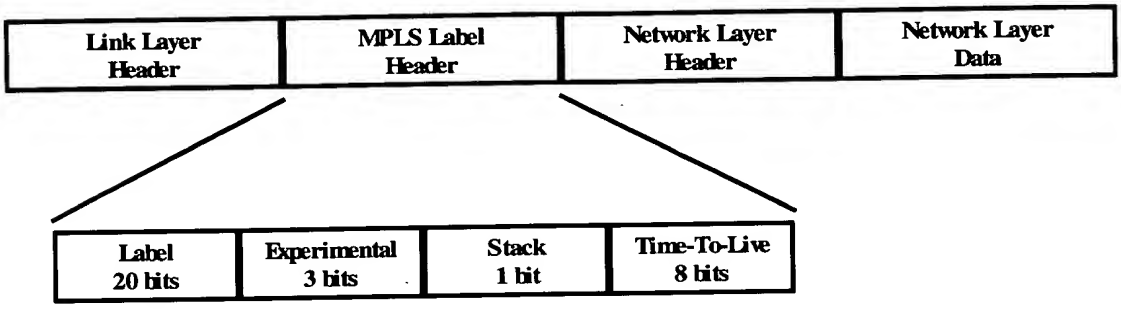


Figure 270

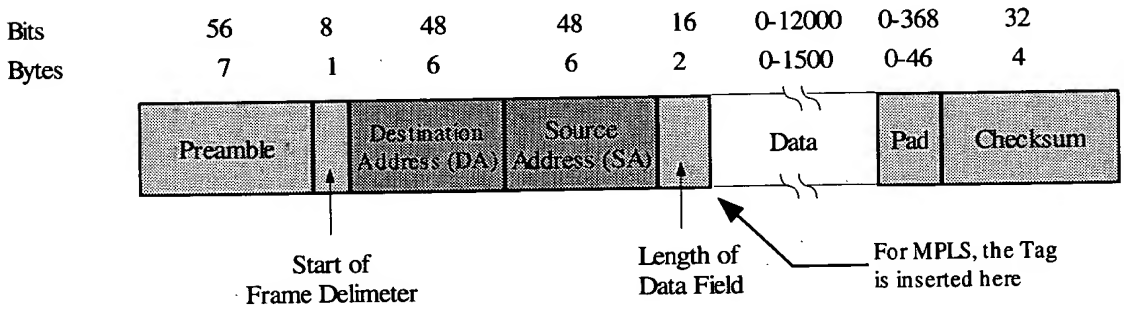


Figure 271

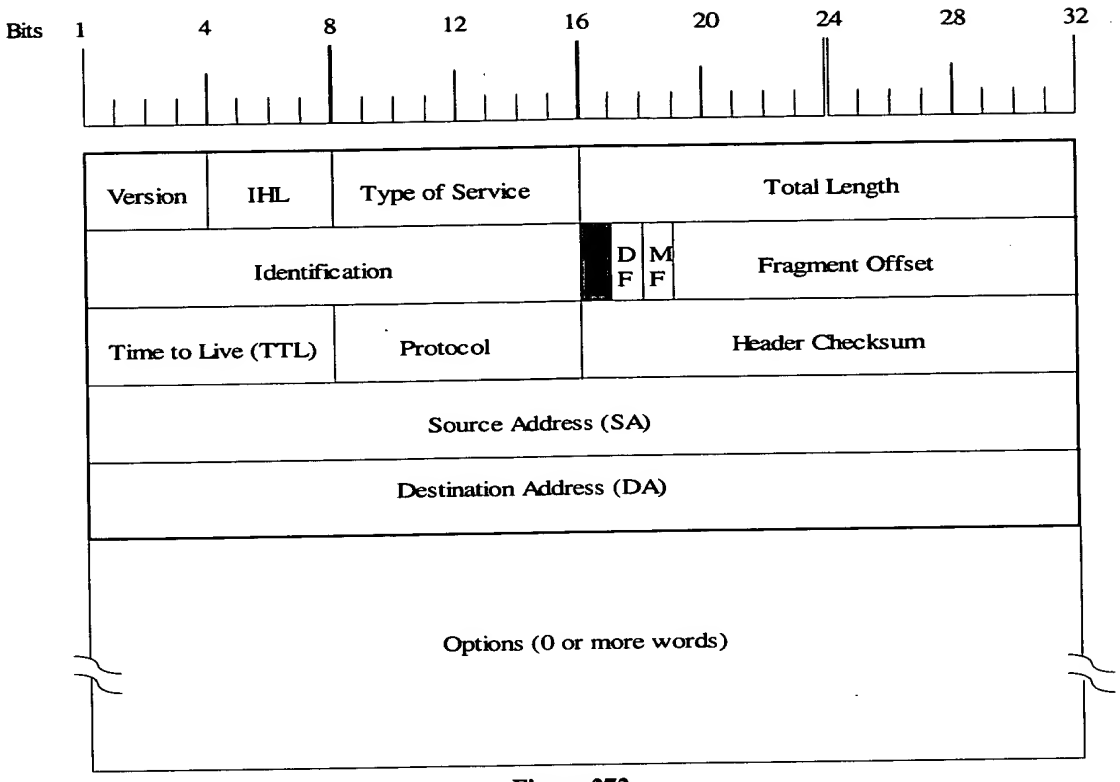
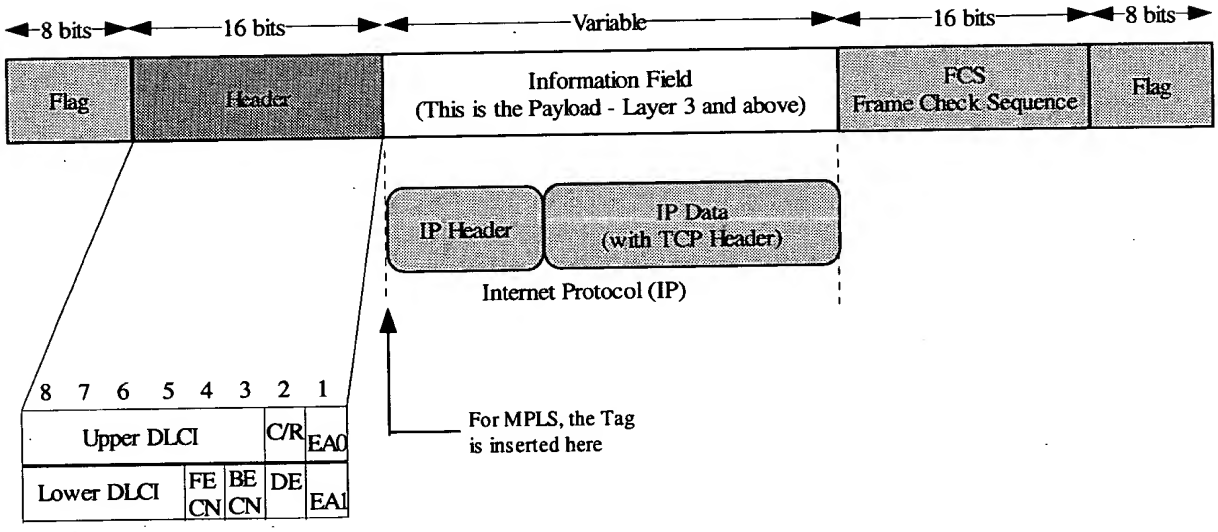


Figure 272



DLCI = Data Link Connection Identifier (10 bits)
C/R = command / response bit
EA = Extended Address Bit
FECN = Forward Explicit Congestion Notification Bit
BECN = Backward Explicit Congestion Notification bit
DE = Discard Eligibility bit

Figure 273

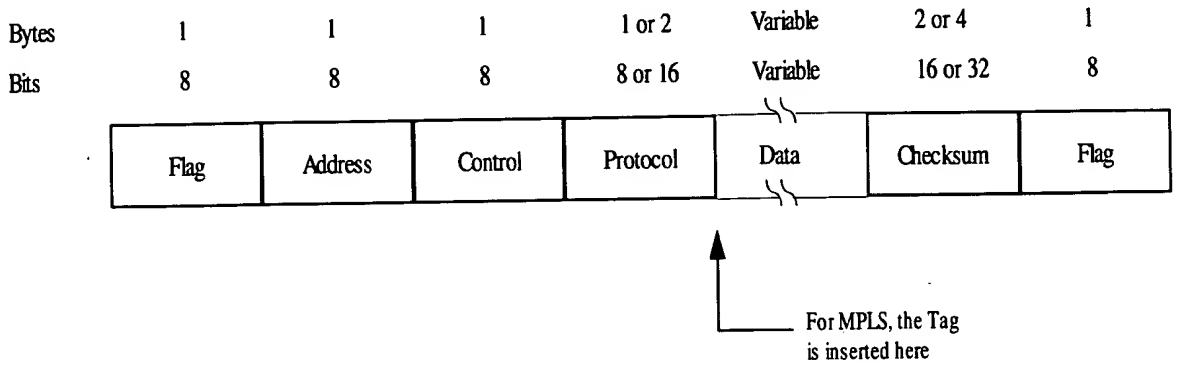


Figure 274

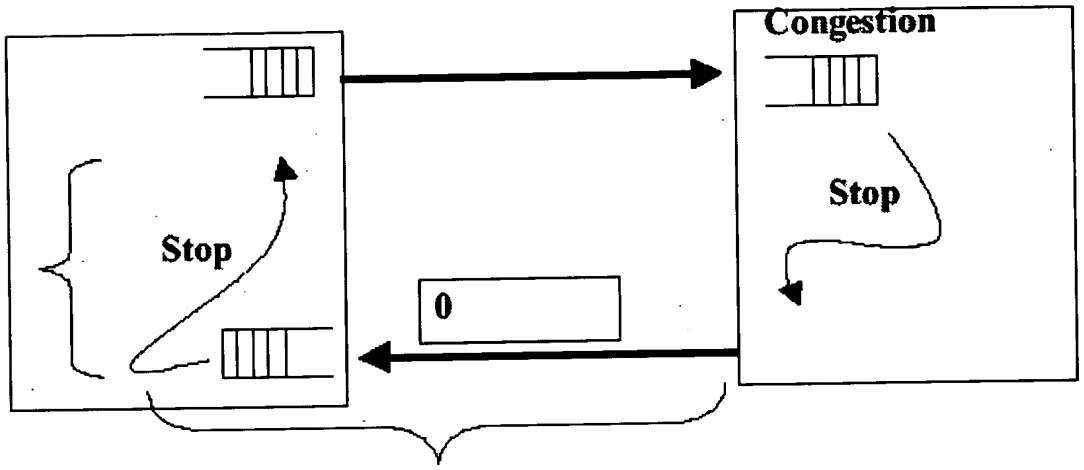


Figure 275

Flow Control Entry Byte Number	Bit Position															
	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
2/3	Class								FC Entry Type		C *	P *	Speed			
4/5	P	CSIX	Destination Address													

Figure 276

Flow Control Function	Flow Control CFrame Fields generated by Fabric			
	CLASS	C*	P*	DA
Traffic TYPE = Unicast				
Adjust speed for unicast traffic for a specific destination (for all classes)	N/A	1	0	DA data
Adjust speed for unicast traffic for a specific destination and specific class only	Class data	0	0	DA data
Adjust speed for all unicast traffic for all destinations and all classes	N/A	1	1	N/A
Adjust speed for all unicast traffic for all destinations and specific class only	Class data	0	1	N/A
Traffic TYPE = Multicast (does not affect broadcast traffic)				
Adjust speed for all multicast traffic for all classes	N/A	1	N/A	N/A
Adjust speed for all multicast traffic for a specific class only	Class data	0	N/A	N/A
Traffic TYPE = Broadcast (does not affect multicast traffic)				
Adjust speed for all broadcast traffic for all classes	N/A	1	N/A	N/A
Adjust speed for all broadcast traffic for a specific class only	Class data	0	N/A	N/A
Traffic TYPE = All (Unicast/Multicast/Broadcast)				
Adjust speed for all traffic types and all classes	N/A	1	N/A	N/A
Adjust speed for all traffic types for a specific class only	Class data	0	N/A	N/A

1) The SPEED field is valid for all of the above function
2) All fields being N/A should be set to all "0" by the Fabric

Figure 277

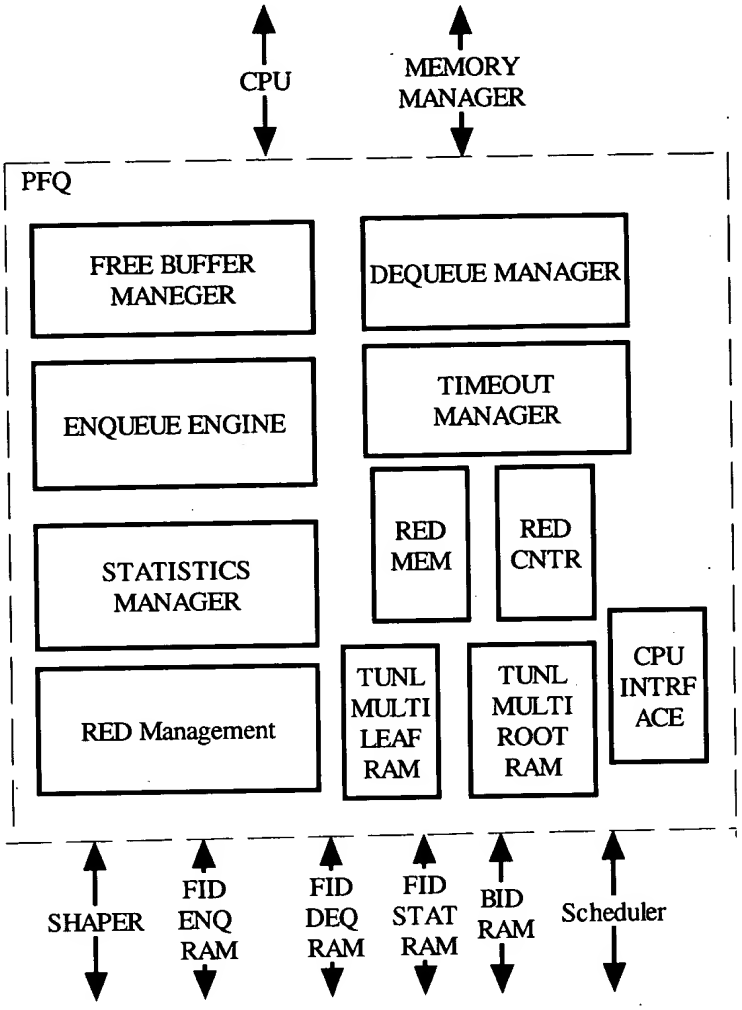
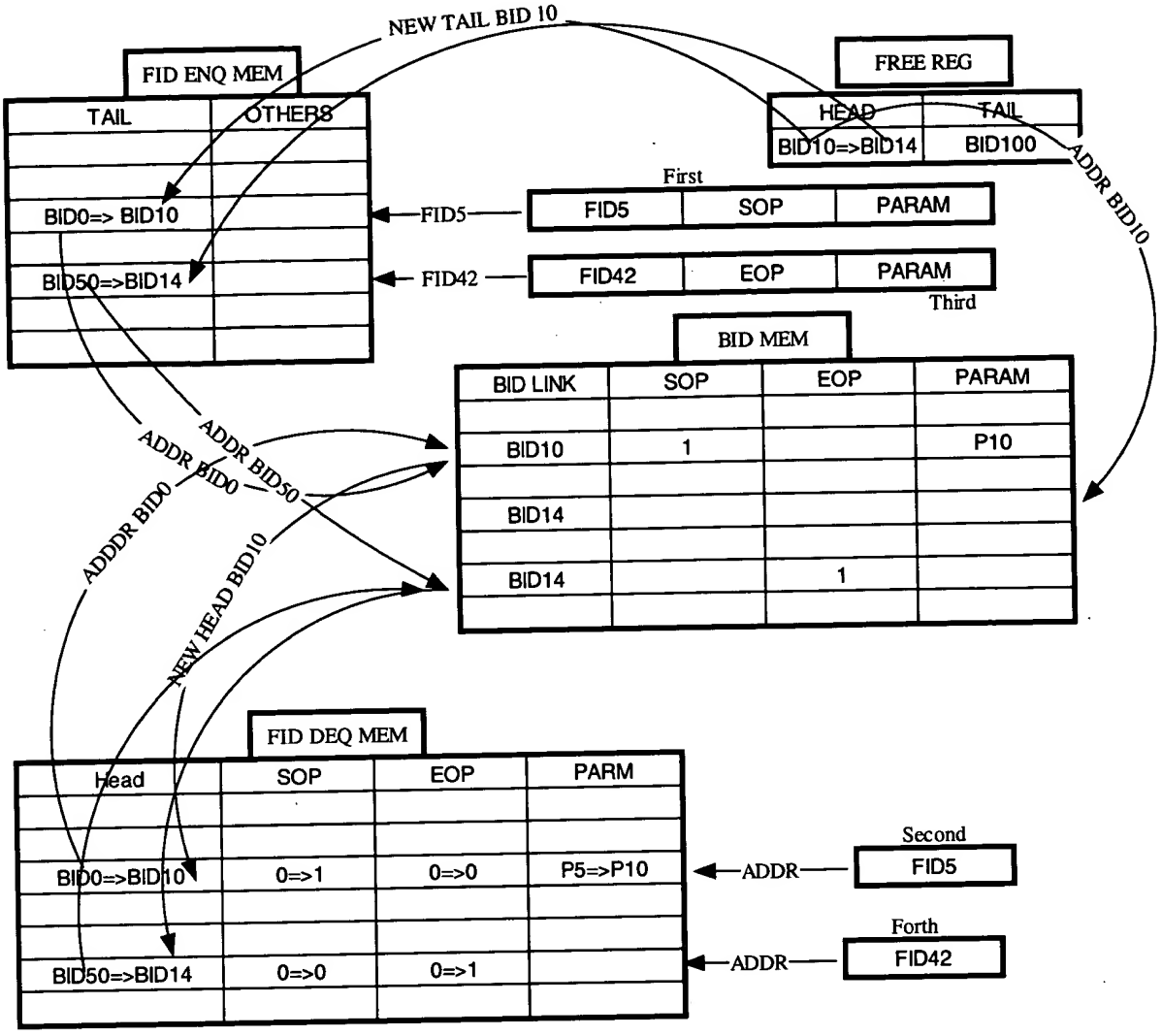

CPU
MEMORY MANAGER
PFQ
FREE BUFFER MANEGER
DEQUEUE MANAGER
ENQUEUE ENGINE
TIMEOUT MANAGER
RED MEM
RED CNTR
STATISTICS MANAGER
CPU INTRF ACE
RED Management
TUNL MULTI LEAF RAM
TUNL MULTI ROOT RAM
SHAPER
FID ENQ RAM
FID DEQ RAM
FID STAT RAM
BID RAM
Scheduler

Figure 278


NEW TAIL BID 10
FID ENQ MEM
TAIL
OTHERS
BID0=> BID10
BID50=>BID14
FREE REG
HEAD
TAIL
BID10=>BID14
BID100
ADDR BID10
First
FID5
SOP
PARAM
FID42
EOP
PARAM
Third
BID MEM
BID LINK
SOP
EOP
PARAM
BID10
1
P10
BID14
1
ADDR BID50
ADDR BID0
ADDDR BID0
NEW HEAD BID10
FID DEQ MEM
Head
SOP
EOP
PARM
BID0=>BID10
0=>1
0=>0
P5=>P10
BID50=>BID14
0=>0
0=>1
Second
ADDR
FID5
Forth
ADDR
FID42

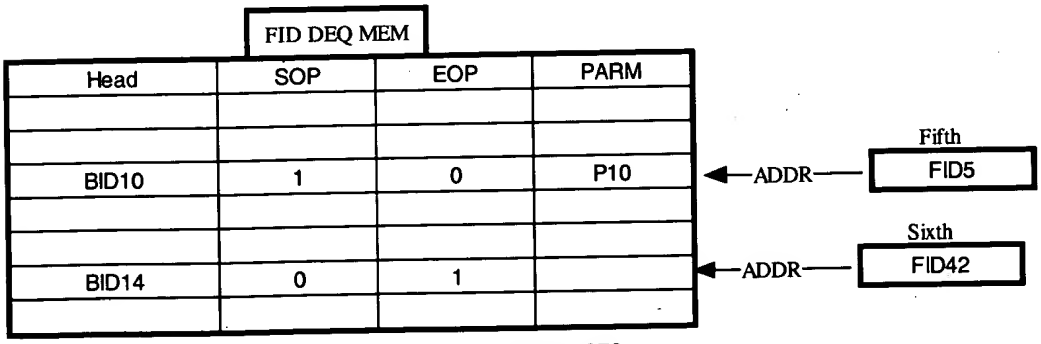

FID DEQ MEM
Head
SOP
EOP
PARM
BID10
1
0
P10
BID14
0
1
Fifth
ADDR
FID5
Sixth
ADDR
FID42

Figure 279

FID	ACT	EOP	SOP	CURRENTLINK	NEW LINK
FID5	ENQ		BIDMEM	BID0	BID10
FID5	DEQ		DEQMEM	BID0	BID10
FID42	ENQ	BIDMEM		BID50	BID14
FID42	DEQ	DEQMEM		BID50	BID14

Figure 280

ENQUEUE INTERNAL BLOCK DIAGRAM

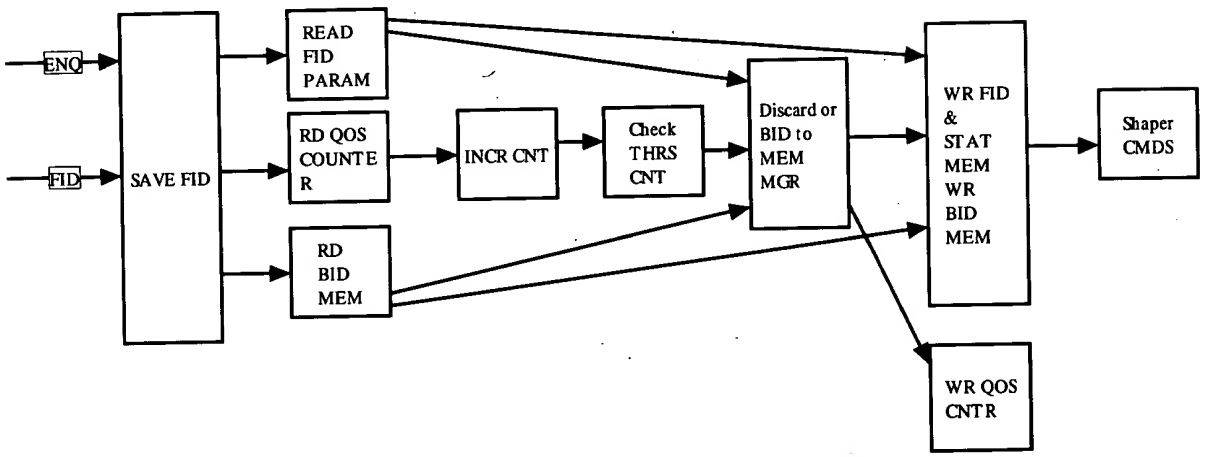


Figure 281

DEQUEUE INTERNAL BLOCK DIAGRAM

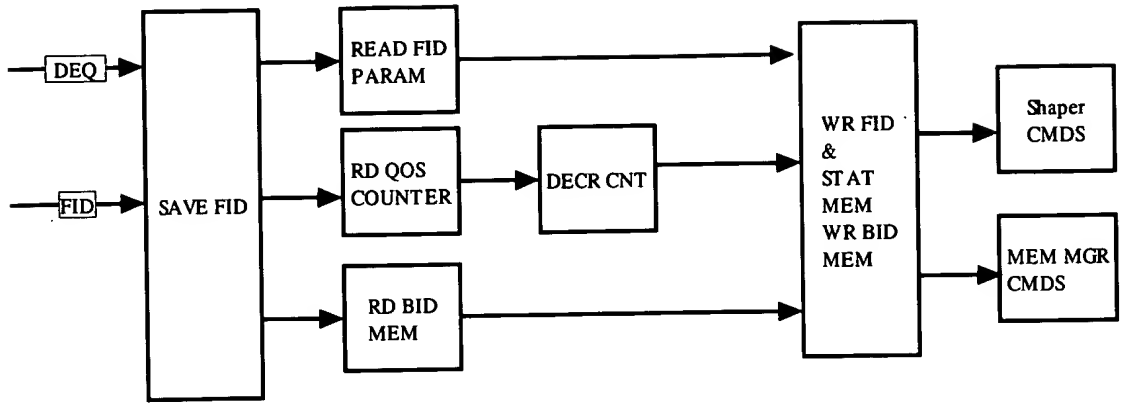


Figure 282

FREE BUFFER INTERNAL BLOCK DIAGRAM

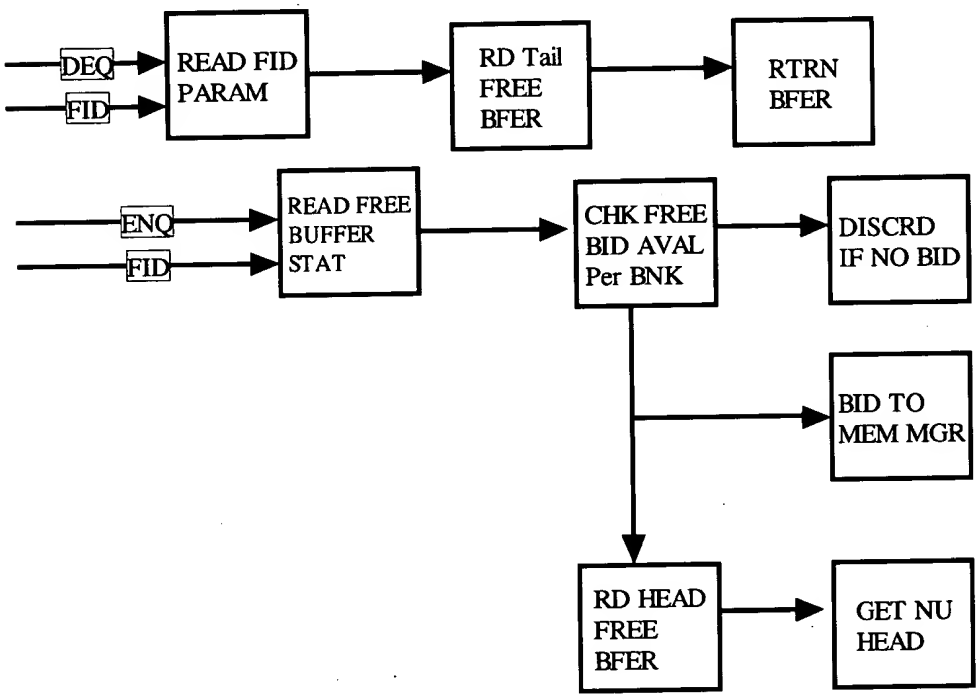


Figure 283

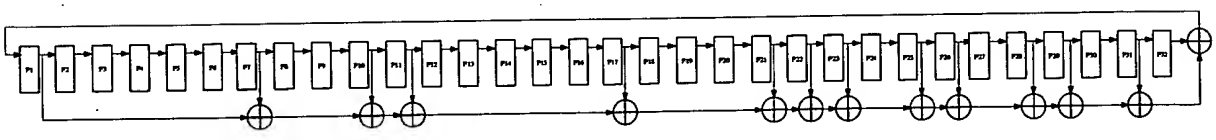
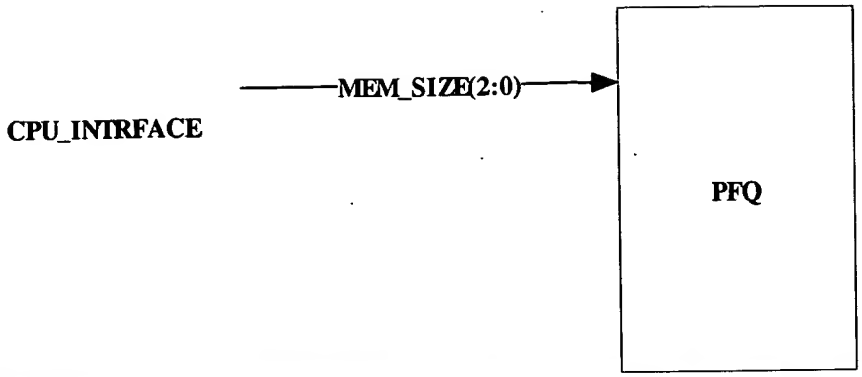


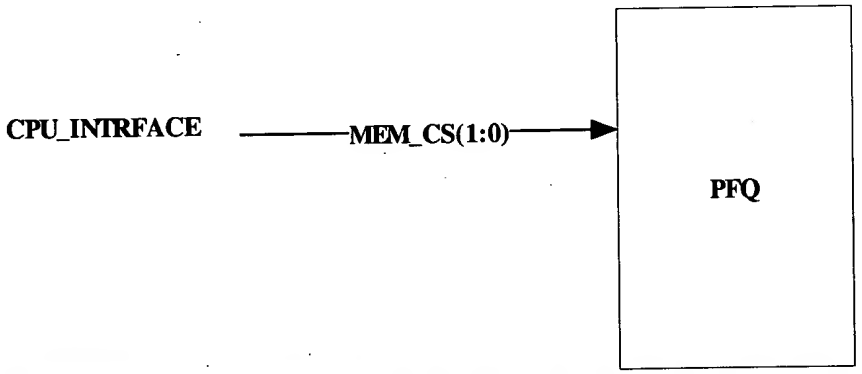
Figure 284



The default value for the number of BIDS is "001" which corresponds to 2 MEG BIDS.

MEM_SIZE INPUT VALUES	NUMBER_BIDS
000	1 M
001	2 M
010	3 M
011	4 M
100	5 M
101	6 M
110	7 M
111	8 M

Figure 285



The default value for chip select is "00".

MEM_CS	ADRESS_BITS
00	20:19
01	21:20
10	22:21
11	Not Used

Figure 286

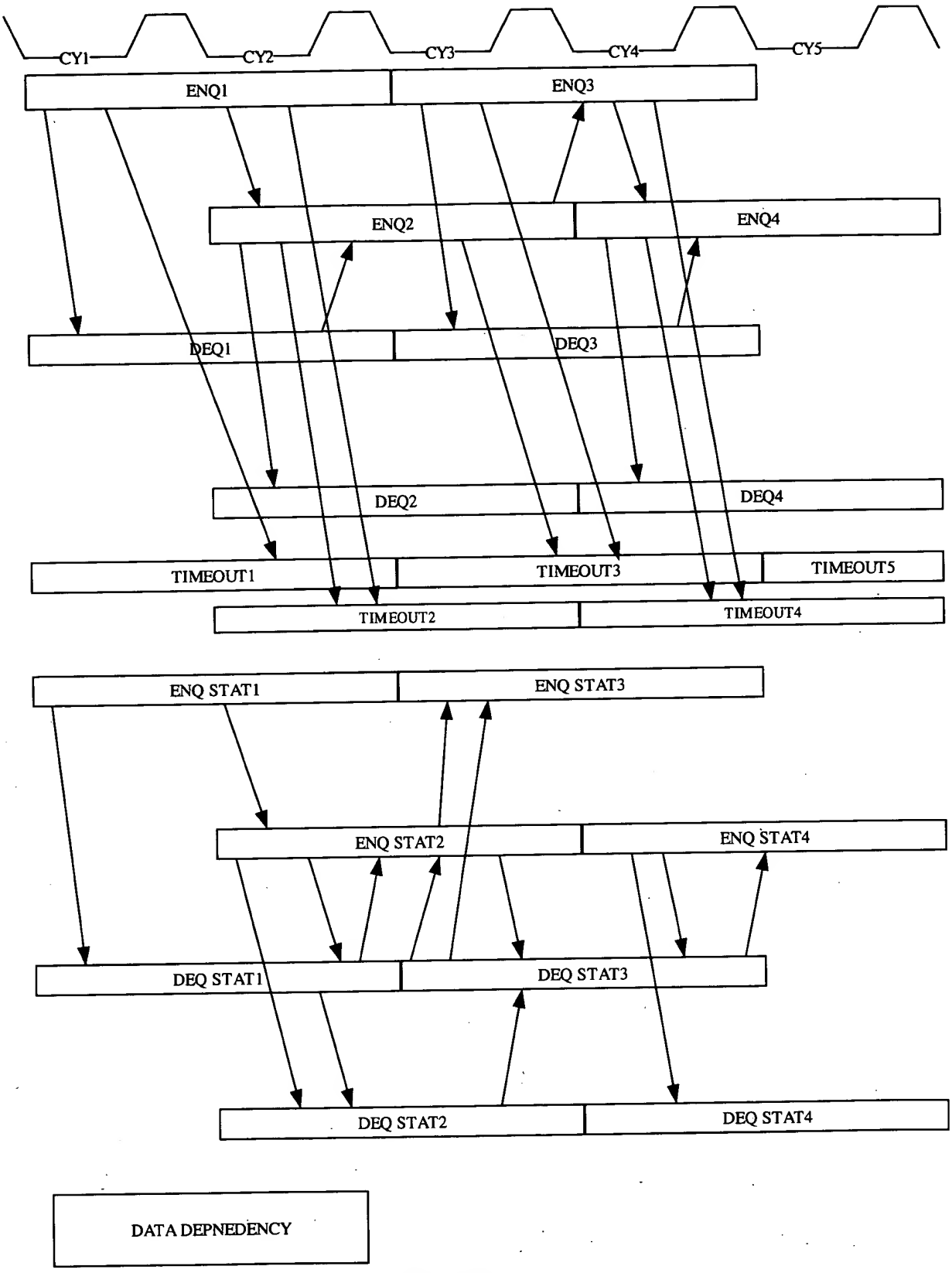

CY1
CY2
CY3
CY4
CY5
ENQ1
ENQ3
ENQ2
ENQ4
DEQ1
DEQ3
DEQ2
DEQ4
TIMEOUT1
TIMEOUT3
TIMEOUT5
TIMEOUT2
TIMEOUT4
ENQ STAT1
ENQ STAT3
ENQ STAT2
ENQ STAT4
DEQ STAT1
DEQ STAT3
DEQ STAT2
DEQ STAT4
DATA DEPNEDENCY

Figure 287

NAME	NO. OF BITS	BITS RANGE	WR	DESCRIPTION
W_q	4	3:0	S	It is the queue weight. An initial value of 0.002 is used. It should not be large in order to filter transient congestions. It represents "N" and W_q is equal to "0.00N"
MAX_p	4	7:4	S	It is the maximum value for a probability "P_b" of discard. An initial value of 1/50 is used. It should be less than 0.1. It represents "N" and MAX_p is equal to "0.0N"
MIN_{th}	16	23:8	S	It is the minimum threshold for queue. If the traffic is bur sty, then the value has to be fairly large.
MAX_{th}	16	39:24	S	It is the maximum threshold for queue. The value determines the maximum average delay in the device. An initial value of 3 * MIN_{th} is used.
RANDOM SEL	8	47:40	S	Each bit selects the nibble of the random number generator that will be used in RED calculations

Figure 288

NAME	OP	BITS RANGE	DESCRIPTION
BYTE_ENABLE	R/W	47:0	Only one byte enable is used for both read and write operations.

Figure 289

NAME	NO. OF BITS	BITS RANGE	WR	DESCRIPTION
Q Size	16	15:0	H	The size of the queue in cells. It is incremented on enqueue with EOP with the number of cells. It is decremented by one on every dequeue operation.
PKT DISCD OTHERS	8	23:16	H	It is incremented once per packet discard when an external command has been received to discard the FID. Bits 15:8 are in the second location.
PACKET CNT	32	55:24	H	It is incremented once per EOP.
PKT DISCD TIMEOUT	16	71:56	H	It is incremented once per packet discard when FID has internally been timeout.

Figure 290

NAME	NO. OF BITS	BITS RANGE	WR	DESCRIPTION
OUTPUT PORT NUMBER	6	5:0	S	It is the output port number that the FID will be transmitted on.
RSVD	3	8:6		Not Used
TOTAL CELL CNT	39	47:9	H	When EOP then the cell count of the packet is added to this counter.
PKT DISCD RED MGMT	16	63:48	H	IT IS INCREMENTED ONCE PER PACKET DISCARD WHEN QUEUE DEPTH COUNTERS PER THE OUTPUT PORT AND CLASS OF THAT FID HAS BEEN EXCEEDED.
PKT DISCD OTHERS	8	71:64	H	It is incremented once per packet discard when an external command has been received to discard the FID.

Figure 291

NAME	OP	BITS RANGE	DESCRIPTION
BYTE_ENABLE_0	R/W	8:0	It is used for both read and writes operations.
BYTE_ENABLE_1	R/W	71:9	It is used for both read and writes operations except when writing the Output port number.

Figure 292

NAME	NO BITS	RANGE	WR	DESCRIPTION
BID HEAD	23	22:0	H	The head is an address to memory. It will point to a buffer. Therefore, it is the Buffer ID head pointer. It is the first buffer pointer that has been enqueued for this FID and the first one to be dequeued. If it is NULL then the queue is empty.
CLASS	3	25:23	H	Class of FID
RSVD	2	27:26		Not Used
HD OAM	1	28	H	OAM BIT
HD EOP PKT	1	29	H	If set, then the head BID is the EOP one for packets.
HD SOP PKT	1	30	H	If set, then the head BID is the SOP one for packets.
HD EFCI	1	31	H	EFCI BIT
FID TYPE	4	35:32	H	The type represents the actions that the device will take in regards to this FID. It is not used in the per flow engine. It will be send to the memory manager. The type will be written with the head pointer after the queue has been empty.

Figure 293

NAME	OP	BITS RANGE	DESCRIPTION
BYTE_ENABLE	R/W	35:0	Only one byte enable is used for both read and write operations.

Figure 294

NAME	NO BITS	RANGE	WR	DESCRIPTION
BID TAIL	23	22:0	H	The tail is an address to memory. It will point to a buffer. Therefore, it is the Buffer ID tail pointer. It is the last buffer pointer that has been enqueued for this FID and the last one to be dequeued. If it is Null, then the queue is empty
CLP	1	23	H	
RSVD	3	26:24		
DROP PORT	1	27	H	Drop Packet because of port parameter
RED/CLASS DROP	1	28	H	Drop the cell until EOP then recheck the RED or CLASS algorithm
TTL	2	30:29	H	When higher than 2, discard & de activate the FID. Reset when dequeue, shaping or empty.
Q_{TIME}	20	50:31	H	The start of queue idle time
DROP Timeout	1	51	H	Drop until next SOP. Keep the queue free of packet fragments because of time out discards.
DROP & discard	1	52	H	Drop until next EOP. Discard last packet fragment.
DROP FREE THRES	1	53	H	Drop until next SOP.
RSVD	10	61:54		
RED ASSOC	10	71:62	S	RED Association. This is set with a setup connection command. It should never be overwritten by the logic. When queue is empty, the byte enable has to be used to write null in the tail pointer.

Figure 295

NAME	NO BITS	RANGE	WR	DESCRIPTION
AVG	16	15:0	H	The average size of the queue
BID PRV PKT TAIL	23	38:16	H	It is the previous packets tail BID. It will point to the current packet HEAD BID. It is saved on EOP.
Tail PKT CELL CNT	10	48:39	H	It represents the number of cells in the tail packet that is being enqueued.
RSVD	6	54:49		Not Used
COUNT	16	70:55	H	The number of not discarded packets that have arrived since last RED discard. It is reset on the next RED discard
RSVD	1	71		Not Used

Figure 296

NAME	OP	BITS RANGE	DESCRIPTION
BYTE_ENABLE _0	R/W	26:0	Asserted for all read operation and for all write operations except when writing RED ASSOCIATION.
BYTE_ENABLE _1	R/W	53:27	Asserted for all read operation and for write operations, except when the dequeue engine is updating the Tail pointer to "NULL" on empty queue or when the RED ASSOCIACTION.
BYTE_ENABLE _2	R/W	71:54	Asserted for all read operation and for write operations, except when the dequeue engine is updating the Tail pointer to "NULL" on empty queue.

Figure 297

NAME	NO BITS	RANGE	WR	DESCRIPTION
BID LINK	23	22:0	H	This is the ID of the next buffer that linked to a specific FID. This is a buffer ID that is linked on an FID queue. Also, it can be a BID that is linked on the Free buffer List.
EOP PKT	1	23	H	This is the END OF PACKET of packet indication for the corresponding BID. The "BID LINK" is the packet cell tail buffer ID. The EOP belongs to the BID link.
SOP PKT	1	24	H	This is the Start OF PACKET of packet indication for the corresponding BID. The "BID LINK" is the packet cell tail buffer ID. The SOP belongs to the BID link.
EFCI	1	25	H	It is EFCI pass through Bit
OAM	1	26	H	It is OAM pass through Bit
DEQ CNT	9	35:27	H	Total number of de-queues before freeing the BID. It is used in multicast. It has to be decremented every time the BID is de-queued. The count belongs to the BID that is the address of this location.

Figure 298

NAME	OP	BITS RANGE	DESCRIPTION
BYTE_ENABLE	R/W	35:0	Only one byte enable is used for both read and write operations.

Figure 299

NAME	NO BITS	RANGE	WR	DESCRIPTION
BID HEAD	23	22:0	H	The head is an address to memory. It will point to a buffer. Therefore, it is the Buffer ID head pointer. It is the first buffer pointer that has been enqueued for this FID and the first one to be dequeued. If it is NULL then the queue is empty.
CLASS	3	25:23	H	Class of FID
RSVD	1	26		Not Used
REL	1	27	H	When set, release the buffer. The count is zero for the BID.
HD OAM	1	28	H	OAM BIT
HD EOP PKT	1	29	H	If set, then the head BID is the EOP one for packets.
HD SOP PKT	1	30	H	If set, then the head BID is the SOP one for packets.
HD EFCI	1	31	H	EFCI BIT
FID TYPE	4	35:32	H	The type represents the actions that the device will take in regards to this FID. It is not used in the per flow engine. It will be send to the memory manager.

Figure 300

NAME	OP	BITS RANGE	DESCRIPTION
BYTE_ENABLE	R/W	35:0	Only one byte enable is used for both read and write operations.

Figure 301

NAME	NO BITS	RANGE	WR	DESCRIPTION
BID TAIL	23	22:0	H	The tail is an address to memory. It will point to a buffer. Therefore, it is the Buffer ID tail pointer. It is the last buffer pointer that has been enqueued for this FID and the last one to be dequeued. If it is Null, then the queue is empty
CLP	1	23	H	
RSVD	3	26:24		
DROP PORT	1	27	H	Drop Packet because of port parameter
RED /CLASS DROP	1	28	H	Drop the cell until EOP then recheck the RED algorithm
TTL	2	30:29	H	When higher than 2, discard & de activate the FID. Reset when dequeue, shaping or empty.
Q-TIME	20	50:31	H	The start of queue idle time
DROP Timeout	1	51	H	Drop until next SOP. Keep the queue free of packet fragments because of time out discards.
DROP & discard	1	52	H	Drop until next EOP. Discard last packet fragment.
DROP FREE THRES	1	53	H	Drop until next SOP.
RSVD	10	61:54		
RED ASSOC	10	71:62	S	RED Association. This is set with a setup connection command. It should never be overwritten by the logic. When queue is empty, the byte enable has to be used to write null in the tail pointer.
RSVD	10	81:72		

Figure 302

NAME	NO BITS	RANGE	WR	DESCRIPTION
AVG	16	15:0	H	The average size of the queue
BID PRV PKT TAIL	23	38:16	H	It is the previous packets tail BID. It will point to the current packet HEAD BID. It is saved on EOP.
Tail PKT CELL CNT	10	48:39	H	It represents the number of cells in the head packet. The head packet is enqueued first
RSVD	6	54:49		Not Used
COUNT	16	70:55	H	The number of not discarded packets that has arrived since last discard
RSVD	1	71		Not Used
NXT FID MULTI	10	81:72	S	It represents the next FID member of the multicast FID

Figure 303

BYTE_ENABLE _0	R/W	26:0	Asserted for all read operation and for all write operations except when writing RED ASSOCIATION or NEXT TUN.
BYTE_ENABLE _1	R/W	53:27	Asserted for all read operation and for write operations, except when the dequeue engine is updating the Tail pointer to "NULL" on empty queue or when the RED ASSOCIACTION or NEXT TUN.
BYTE_ENABLE _2	R/W	71:54	Asserted for all read operation and for write operations, except when the dequeue engine is updating the Tail pointer to "NULL" on empty queue or NEXT TUN.
BYTE_ENABLE _3	R/W	81:72	Asserted for all read operation and for write operations, except when the dequeue engine is updating the Tail pointer to "NULL" on empty queue or when the RED ASSOCIACTION.

Figure 304

NAME	NO BITS	RANGE	WR	DESCRIPTION
FID MULTI HEAD	10	9:0	H/S	It is the current head FID in the MULTICAST flow. It will be updated when the EOP is reached.
FID MULTI TAIL	10	19:10	S	The tail is pointing to the last flow ID in the Multicast flow ID.
COUNT	9	28:20	S	The count that represents the members of the multicast group.
Valid	1	29	S	The entries are valid

Figure 305

NAME	NO BITS	RANGE	WR	DESCRIPTION
BID HEAD	23	22:0	H	The head is an address to memory. It will point to a buffer. Therefore, it is the Buffer ID head pointer. It is the first buffer pointer that has been enqueued for this FID and the first one to be dequeued. If it is NULL then the queue is empty.
CLASS	3	25:23	H	Class of FID
RSVD	5	27:26		Not Used
HD OAM	1	28	H	OAM BIT
HD EOP PKT	1	29	H	If set, then the head BID is the EOP one for packets.
HD SOP PKT	1	30	H	If set, then the head BID is the SOP one for packets.
HD EFCI	1	31	H	EFCI BIT
FID TYPE	4	35:32	H	The type represents the actions that the device will take in regards to this FID. It is not used in the per flow engine. It will be send to the memory manager.
FID TUN ROOT	10	45:36	S	It is the root of the tunneled FID.
NXT FID TUN	10	55:46	S	It represents the next FID member of the Tunneled FID

Figure 306

NAME	NO BITS	RANGE	WR	DESCRIPTION
BID TAIL	23	22:0	H	The tail is an address to memory. It will point to a buffer. Therefore, it is the Buffer ID tail pointer. It is the last buffer pointer that has been enqueued for this FID and the last one to be dequeued. If it is Null, then the queue is empty
CLP	1	23	H	
RSVD	3	26:24		
DROP PORT	1	27	H	Drop Packet because of port parameter
RED/CLASS DROP	1	28	H	Drop the cell until EOP then recheck the RED algorithm
TTL	2	30:29	H	When higher than 2, discard & de activate the FID. Reset when dequeue, shaping or empty.
Q-TIME	20	50:31	H	The start of queue idle time
DROP Timeout	1	51	H	Drop until next SOP. Keep the queue free of packet fragments because of time out discards.
DROP & discard	1	52	H	Drop until next EOP. Discard last packet fragment.
DROP FREE THRES	1	53	H	Drop until next SOP.
RSVD	10	61:54		
RED ASSOC	10	71:62	S	RED Association. This is set with a setup connection command. It should never be overwritten by the logic. When queue is empty, the byte enable has to be used to write null in the tail pointer.
RSVD	10	81:72		

Figure 307

NAME	NO BITS	RANGE	WR	DESCRIPTION
AVG	16	15:0	H	The average size of the queue
BID PRV PKT TAIL	23	38:16	H	It is the previous packets tail BID. It will point to the current packet HEAD BID. It is saved on EOP.
Tail PKT CELL CNT	10	48:39	H	It represents the number of cells in the tail packet that is being enqueued.
RSVD	6	54:49		Not Used
COUNT	16	70:55	H	The number of not discarded packets that has arrived since last discard
RSVD	1	71		Not Used
FID TUN ROOT	10	81:72	S	It is the root of the tunneled FID.

Figure 308

NAME	OP	BITS RANGE	DESCRIPTION
BYTE_ENABLE _0	R/W	26:0	Asserted for all read operation and for all write operations except when writing RED ASSOCIATION or NEXT TUN.
BYTE_ENABLE _1	R/W	53:27	Asserted for all read operation and for write operations, except when the dequeue engine is updating the Tail pointer to "NULL" on empty queue or when the RED ASSOCIACTION or NEXT TUN.
BYTE_ENABLE _2	R/W	71:54	Asserted for all read operation and for write operations, except when the dequeue engine is updating the Tail pointer to "NULL" on empty queue or NEXT TUN.
BYTE_ENABLE _3	R/W	81:72	Asserted for all read operation and for write operations, except when the dequeue engine is updating the Tail pointer to "NULL" on empty queue or when the RED ASSOCIACTION.

Figure 309

NAME	NO BITS	RANGE	WR	DESCRIPTION
CRNT FID TUN HEAD	10	9:0	H/S	IT IS THE CURRENT FID IN THE TUNNELED FLOW. IT WILL BE UPDATED WHEN THE EOP IS REACHED.
FID TUN TAIL	10	19:10	H/S	IT IS THE LAST FID IN THE LINK.
FID not Empty	1	20	S	If it is set, then the FID is not empty and active.

Figure 310

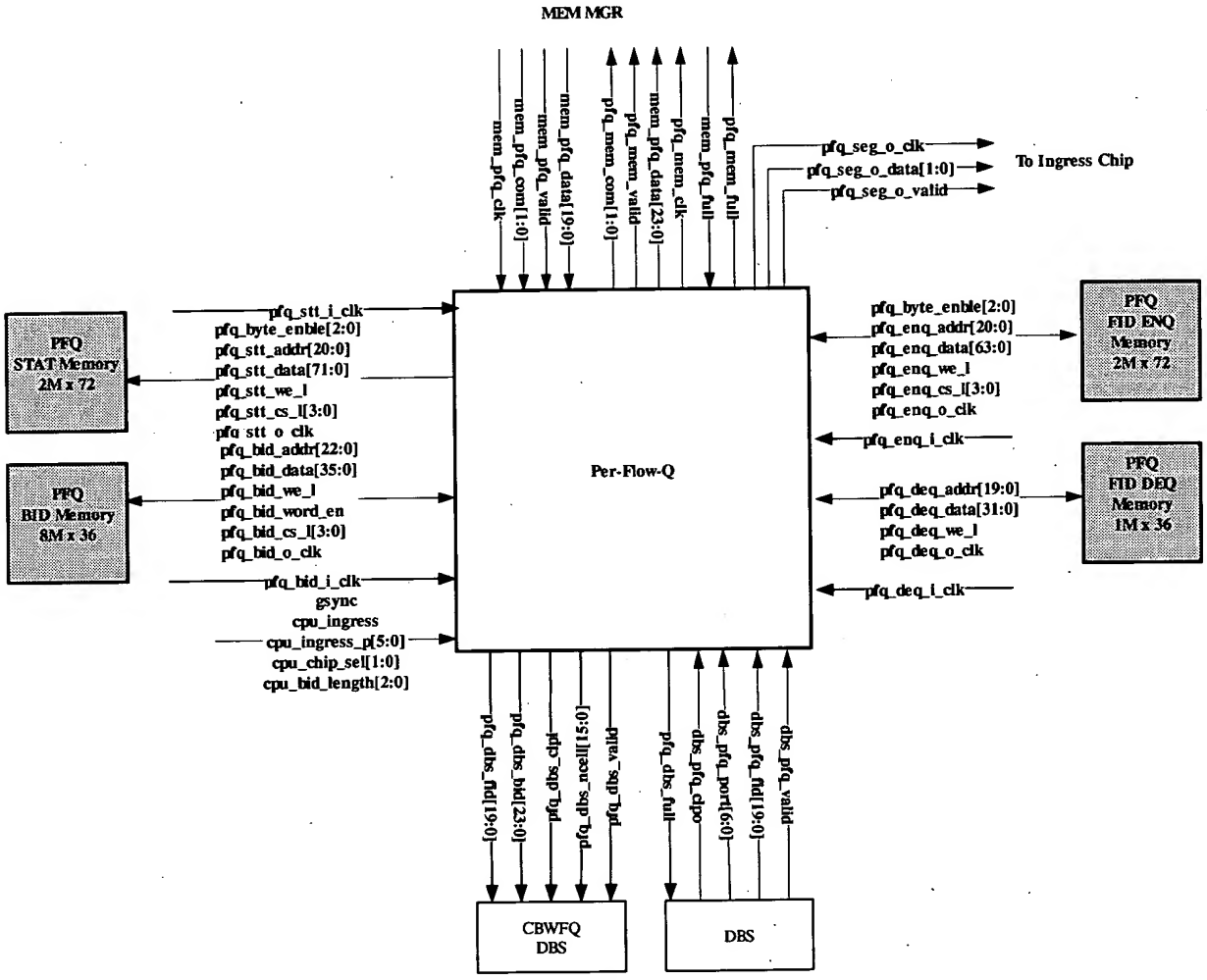


Figure 311

SIGNAL	DESCRIPTION
BID_MEM_SIZE [2:0]	Specifies the BID memory size
BID_MEM_CS [2:0]	Specifies the BID memory Chip Selects with respect to address bits.

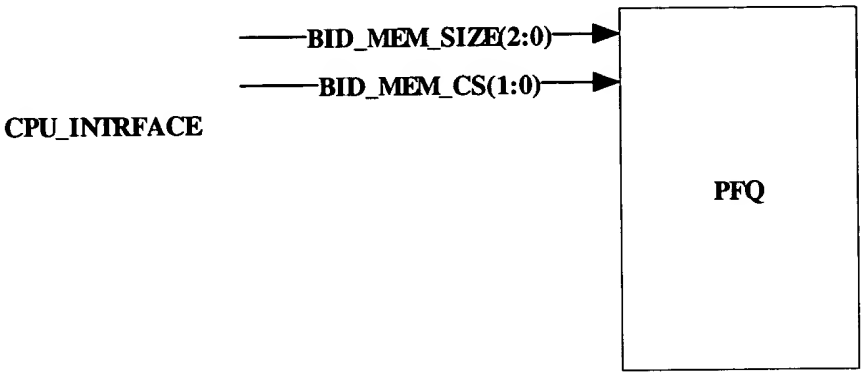
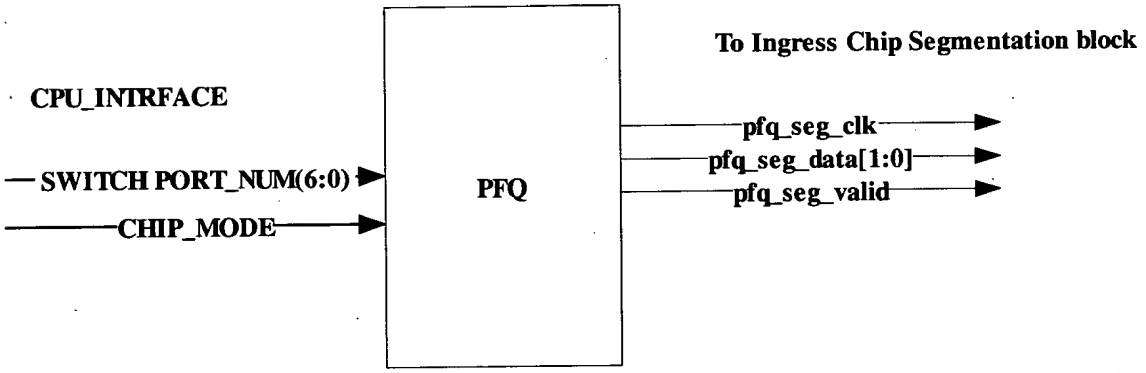


Figure 312



NAME	DIR	VALUE
CHIP_MODE	IN	0 – ingress chip 1 – egress chip 1 –Default
PORT_NUM (6:0)	IN	Switch Device Port Number 0000000 – Default Value

Figure 313

NAME	DIR	VALUE
PFQ_SEG_CLK	OUT	CLOCK
PFQ_SEG_DATA [1:0]	OUT	The transfer takes four clocks CLK DATA1 DATA0 1 6 5 2 4 3 3 2 1 4 0 F/NF
PFQ_SEG_VALID	OUT	Transfer Valid

Figure 314

PFQ_DBS_VALID	1	O	SIGNALS VALID VALUES ON OTHER SIGNALS FROM PFQ TO SHP 000 – NOT VALID COMMAND (IGNORE ALL OTHER FIELDS)
PFQ_DBS_FLOWID	20	O	FLOWID value to the Shaper block
PFQ_DBS_NCELL	10	O	Number of cells for the FID
PFQ_DBS_BID	23	O	Buffer ID/Pkt ID of the fid.
PFQ_SHP_CLPI	1	O	Cell Loss Priority bit, which the SHP modifies according to the Dual Leaky Bucket algorithm.

Figure 315

DBS_PFQ_PORTID	7	I	PORTID FOR THE SCHEDULED FLOWID
DBS_PFQ_FID	20	I	FLOWID that is scheduled now
DBS_PFQ_VALID	1	I	If asserted, the FID_O and PID are valid, A valid FLOWID and PORTID is driven from the scheduler once every 9 clocks of 200MHz
DBS_PFQ_CLPO	1	I	CLP bit of the current cells. This bit is modified by the shaper and is being sent to the PFQ through the scheduler.
PFQ_ DBS_FULL	1	O	It is bypassed from the Memory Manager or generated internally for tunneled flow IDS

Figure 316

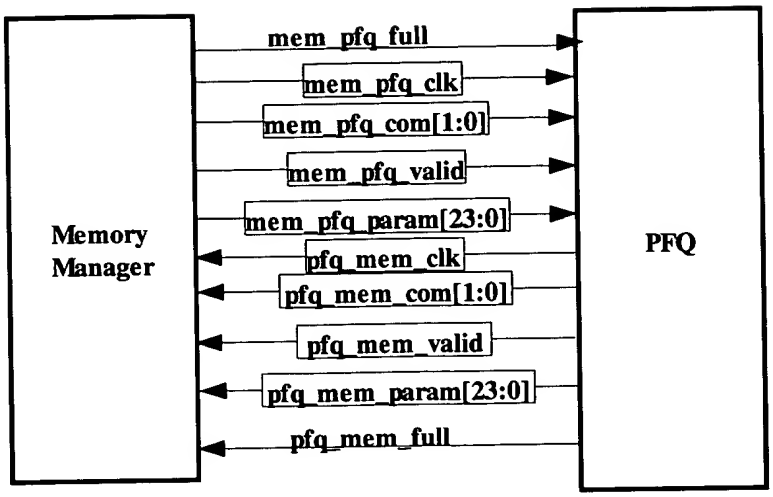


Figure 317

MEM_PFQ_FULL	1	I	MEMORY MANAGER IS FULL
MEM_PFQ_CLK	1	I	Clock
MEM_PFQ_COM_(1:0)	2	I	Command
MEM_PFQ_VALID	1	I	Valid
MEM_PFQ_PARAM	24	I	Parameters

PFQ_MEM_FULL	1	I	PFQ IS FULL
PFQ_MEM_CLK	1	O	Clock
PFQ_MEM_COM_(1:0)	2	O	Command
PFQ_MEM_VALID	1	O	Valid
PFQ_MEM_PARAM	24	O	Parameters

Figure 318

From To	Command	Description	Opcode
MEM -> PFQ	NOP	No command is performed	00
MEM -> PFQ	En-Queue	Enqueue the current cell	01
MEM -> PFQ	Discard	Discard the packet	10
MEM -> PFQ	Release	Release BID	11

Figure 319

PFQ -> MEM	NOP	No command is performed	00
PFQ -> MEM	BID_VALUE	Send a BID for the enqueue command This is a respond to the enqueue command sent from the MEM block.	01
PFQ -> MEM	De-Queue	Dequeue the current cell	10
		Invalid command	11

Figure 320

CMD	OP	CYCLES NEEDED	CYCLES VAULE
01	Enqueue	D1, D2	D1: {4'b0, FID} D2: {12'b0, OAM, EFCI, TYPE [3:0], EOP, SOP, CLP, CLASS [2:0]}
10	Discard	D1	D1: {4'b0, FID}
11	Release	D1	D1: {1'b0, BID}

Figure 321

CMD	OP	CYCLES NEEDED	CYCLES VAULE
01	BID_VALUE	D1	D1: {1'b0, BID}
10	Dequeue	D1, D2, D3	D1: {4'b0, FID} D2: {7'b0, REL, OAM, EFCI, PROTID [6:0], TYPE [3:0], EOP, SOP, CLP} D3: {1'b0, BID}
11	Not Valid	N/A	N/A

Figure 322

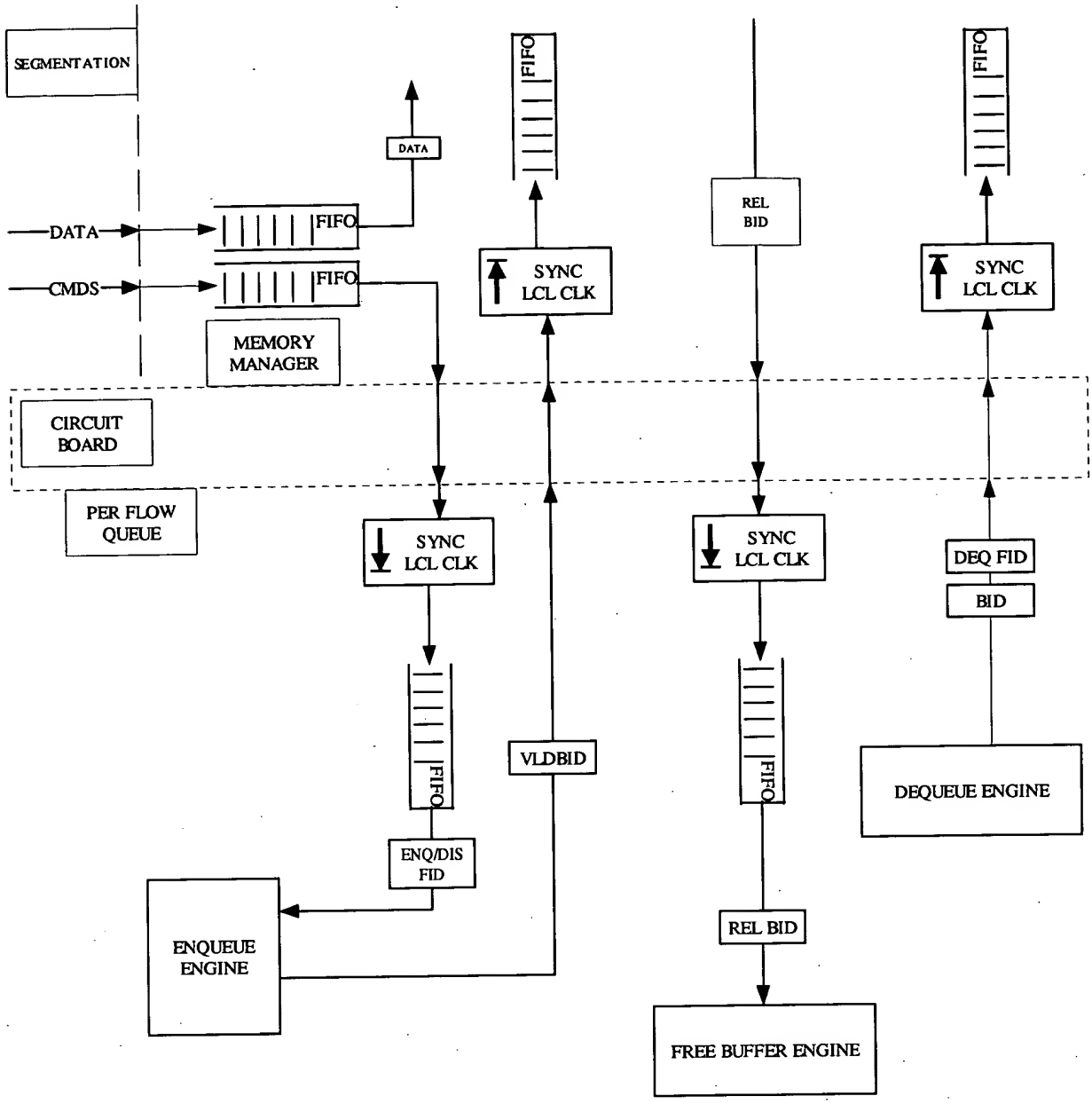

SEGMENTATION
DATA
FIFO
DATA
CMDS
FIFO
MEMORY MANAGER
CIRCUIT BOARD
PER FLOW QUEUE
SYNC LCL CLK
FIFO
ENQ/DIS FID
ENQUEUE ENGINE
FIFO
SYNC LCL CLK
VLDBID
REL BID
SYNC LCL CLK
FIFO
REL BID
FREE BUFFER ENGINE
FIFO
SYNC LCL CLK
DEQ FID
BID
DEQUEUE ENGINE

Figure 323

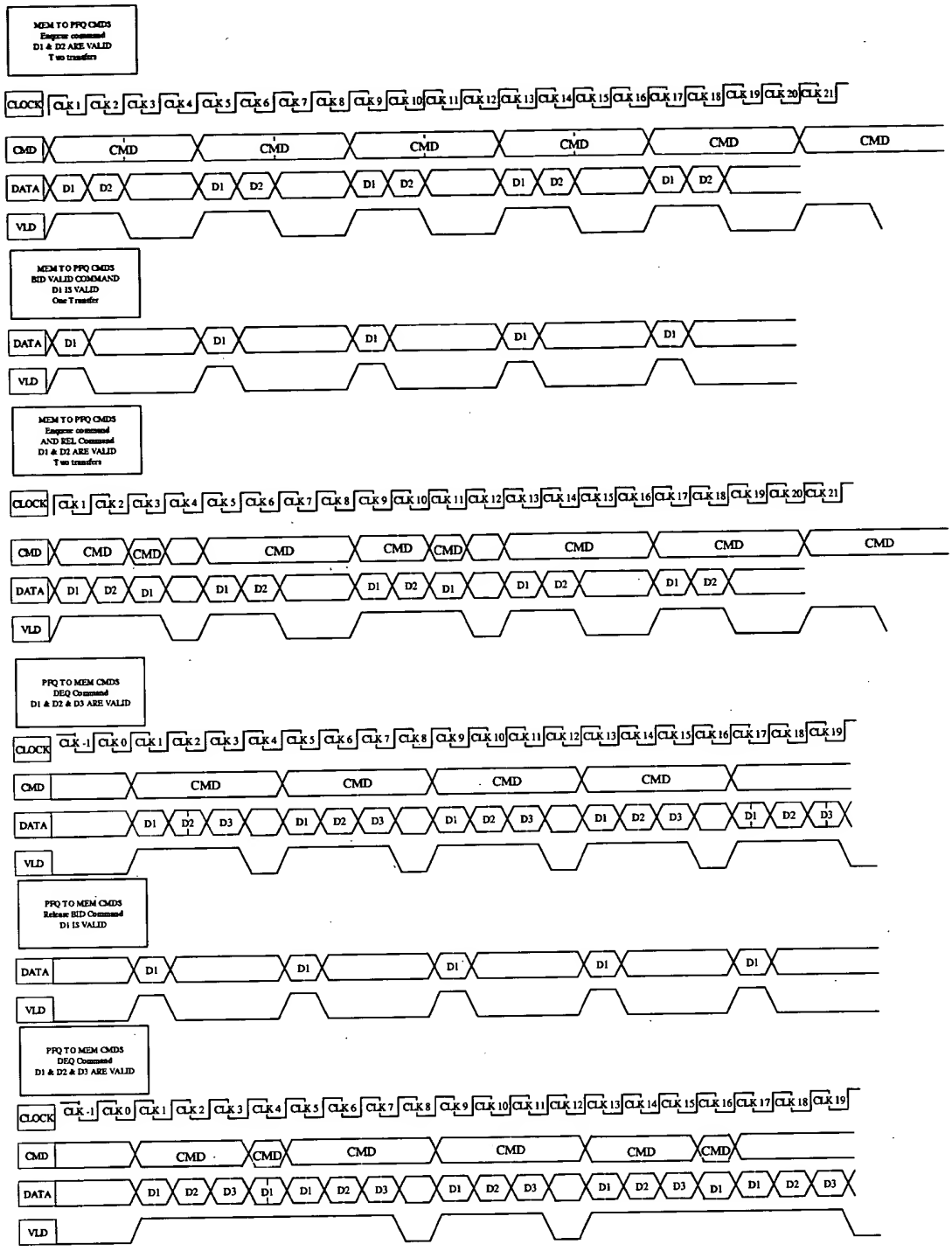

MEM TO PFQ CMDS
Enqcmc command
D1 & D2 ARE VALID
Two transfers
CLOCK
CMD
DATA
VLD
MEM TO PFQ CMDS
BID VALID COMMAND
D1 IS VALID
One Transfer
MEM TO PFQ CMDS
Enqcmc command
AND REL Command
D1 & D2 ARE VALID
Two transfers
PFQ TO MEM CMDS
DEQ Command
D1 & D2 & D3 ARE VALID
PFQ TO MEM CMDS
Release BID Command
D1 IS VALID
PFQ TO MEM CMDS
DEQ Command
D1 & D2 & D3 ARE VALID

Figure 324

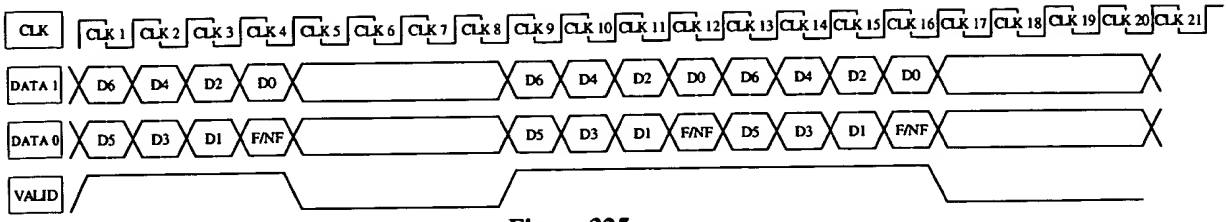


Figure 325

Start address	Last address	Total length
80h	7Fh	64d

Figure 326

Address	Name	Type	Description
0	COM	R/W	[31:28] – Opcode [27:0] – Address, depending on the command. No default value.
1	R0	R/W	General-purpose register. No default value
2	R1	R/W	General-purpose register. No default value
3	R2	R/W	General-purpose register. No default value
4-31	Reserved	NOP	NOT USED
32	TOTAL_FREE_BUFF	R/W	[31:23] – Reserved [22:0] – Total number of Free Buffers Default: 8M – 1 = FFFFE = 1111 1111 1111 1111 1110
33	THRESHOLD_FREE_BUFF	R/W	[31:23] – Reserved [22:0] – Total number of Free Buffers threshold for empty. It has to be less than TOTAL_FREE_BUFFER VALUE Default: 8M – 11H = FFFEE = 1111 1111 1111 1110 1110
34	THRESHOLD_FREE_BUFF_BACK_PRES	R/W	[31:23] – Reserved [22:0] – Total number of Free Buffers threshold for backpressure. It has to be less than THRESHOLD_ FREE_BUFF Default: 8M – 11H = FD8FF = 1111 1101 1000 1111 1111
35	FREE_BUFFS_IN USE	R/W	[31:23] – Reserved [22:0] – Buffers currently in use Loadable counter for testing

36	CONTROL	R/W	[31:4] – Reserved CPU Port blocked [3] if set the CPU port is Blocked. Default value is "1". Enqueue Multicast as Unicast traffic [2] If set the multicast traffic is treated as Unicast. Default value "0". Enable buffer management [1] If set the buffer management is enabled. Default value "0". RED/CLASS [0] If set the buffer management is RED. Default value "0".
37	RED TIME Q-TIME	R/W	[31:25] – Reserved, Counter [24:0] - Count
38	Transmission time "S"	R/W	[31:16] Reserved, [15:0] Typical transmission time for a small packet.
39	FID Memory descriptor	R/W	[31:20] Reserved, [19:0] MASK contiguously used. Number of bits used to access the FID memory. The number of buffers can be less than the available storage. Default value "FFFFF"
40	Timeout Rate	R/W	[31:5] Reserved, [4:0] Timeout rate. It is the number of clocks to skip before the next timeout. Default value is "00000".
41	FREE Buffer Tail	R/W	Free Buffer tail [22:0]
42	FREE Buffer Head	R/W	Free Buffer Head [22:0]
43	OUTPUT PORT BLOCKED [31:0]	R/W	[31:0] Output ports 31 downto 0 statuses. If set, then the port is blocked and discard command with EOP cell is asserted.
44	OUTPUT PORT BLOCKED [63:32}	R/W	[31:0] Output ports 63 downto 32 statuses. If set, then the port is blocked and discard command with EOP cell is asserted.
45	CLASS 0 Threshold	R/W	[23:0] CLASS 0 buffer management Threshold. Default value "000FFF".
46	CLASS 1 Threshold	R/W	[23:0] CLASS 1 buffer management Threshold. Default value "000FFF".
47	CLASS 2 Threshold	R/W	[23:0] CLASS 2 buffer management Thresholds. Default value "000FFF".
48	CLASS 3 Threshold	R/W	[23:0] CLASS 3 buffer management Thresholds. Default value "000FFF".
49	CLASS 4 Threshold	R/W	[23:0] CLASS 4 buffer management Thresholds. Default value "000FFF".
50	CLASS 5 Threshold	R/W	[23:0] CLASS 5 buffer management Thresholds. Default value "000FFF".
51	CLASS 6 Threshold	R/W	[23:0] CLASS 6 buffer management Thresholds. Default value "000FFF".

52	CLASS 7 Threshold	R/W	[23:0] CLASS 7 buffer management Thresholds. Default value "000FFF".
53	CLASS 0 Counter	R/W	[23:0] CLASS 0 buffer management Counter.
54	CLASS 1 Counter	R/W	[23:0] CLASS 1 buffer management Counter.
55	CLASS 2 Counter	R/W	[23:0] CLASS 2 buffer management Counter.
56	CLASS 3 Counter	R/W	[23:0] CLASS 3 buffer management Counter.
57	CLASS 4 Counter	R/W	[23:0] CLASS 4 buffer management Counter.
58	CLASS 5 Counter	R/W	[23:0] CLASS 5 buffer management Counter.
59	CLASS 6 Counter	R/W	[23:0] CLASS 6 buffer management Counter.
60	CLASS 7 Counter	R/W	[23:0] CLASS 7 buffer management Counter.
61	TEST_REG	R/W	Test mode
62 - 63	Reserved	NOP	NOT USED

Figure 327

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0000	Don't Care						
R0	Don't Care							
R1	Don't Care							

Figure 328

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0001	Don't Care		FID_ADDR [20:0]				
R0	R0 [31:0] Data; Written by the PFQ and Read by the CPU							
R1	R1 [31:0] Data; Written by the PFQ and Read by the CPU							
R2	R2 [7:0] Data; Written by the PFQ and Read by the CPU							

Figure 329

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0010	Don't Care		FID_ADDR [20:0]				
R0	R0 [31:0] Data; Written by the CPU							
R1	R1 [31:0] Data; Written by the CPU							
R2	R2 7:0] Data; Written by the CPU							

Figure 330

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0011	Don't Care		FID_ADDR [19:0]				
R0	R0 [31:0] Data; Written by PFQ and Read by CPU							
R1	R0 [3:0] Data; Written by PFQ and Read by CPU							

Figure 331

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0100	Don't Care		FID_ADDR [19:0]				
R0	R0 [31:0] Data; Written by the CPU							
R1	R1 [3:0] Data; Written by the CPU							

Figure 332

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0101	BID_ADDR [22:0]						
R0	R0 [31:0] Data; Written by the PFQ and Read by the CPU							
R1	R1 [3:0]; Written by the PFQ and Read by the CPU							

Figure 333

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0110	BID_ADDR [22:0]						
R0	R0 [31:0] Data; Written by the CPU							
R1	R1 [3:0]; Written by the CPU							

Figure 334

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0111	Don't Care		FID_ADDR [20:0]				
R0	R0 [31:0] Data; Written by the PFQ and Read by the CPU							
R1	R1 [31:0] Data; Written by the PFQ and Read by the CPU							
R2	R2 [7:0] Data; Written by the PFQ and Read by the CPU							

Figure 335

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1000	Don't Care		FID_ADDR [20:0]				
R0	R0 [31:0] Data; Written by the CPU							
R1	R1 [31:0] Data; Written by the CPU							
R2	R2 [7:0] Data; Written by the CPU							

Figure 336

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1001	Don't Care		FID_ADDR [19:0]				
R0	RED ASSOCIATION [9:0] 9:0 PORT [5:0] 15:10							

Figure 337

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1010	Don't Care		FID_ADDR [19:0]				

Figure 338

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1011	Don't Care						
R0	Don't Care							
R1	Don't Care							

Figure 339

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1100	Don't Care						
R0	Don't Care							
R1	Don't Care							

Figure 340

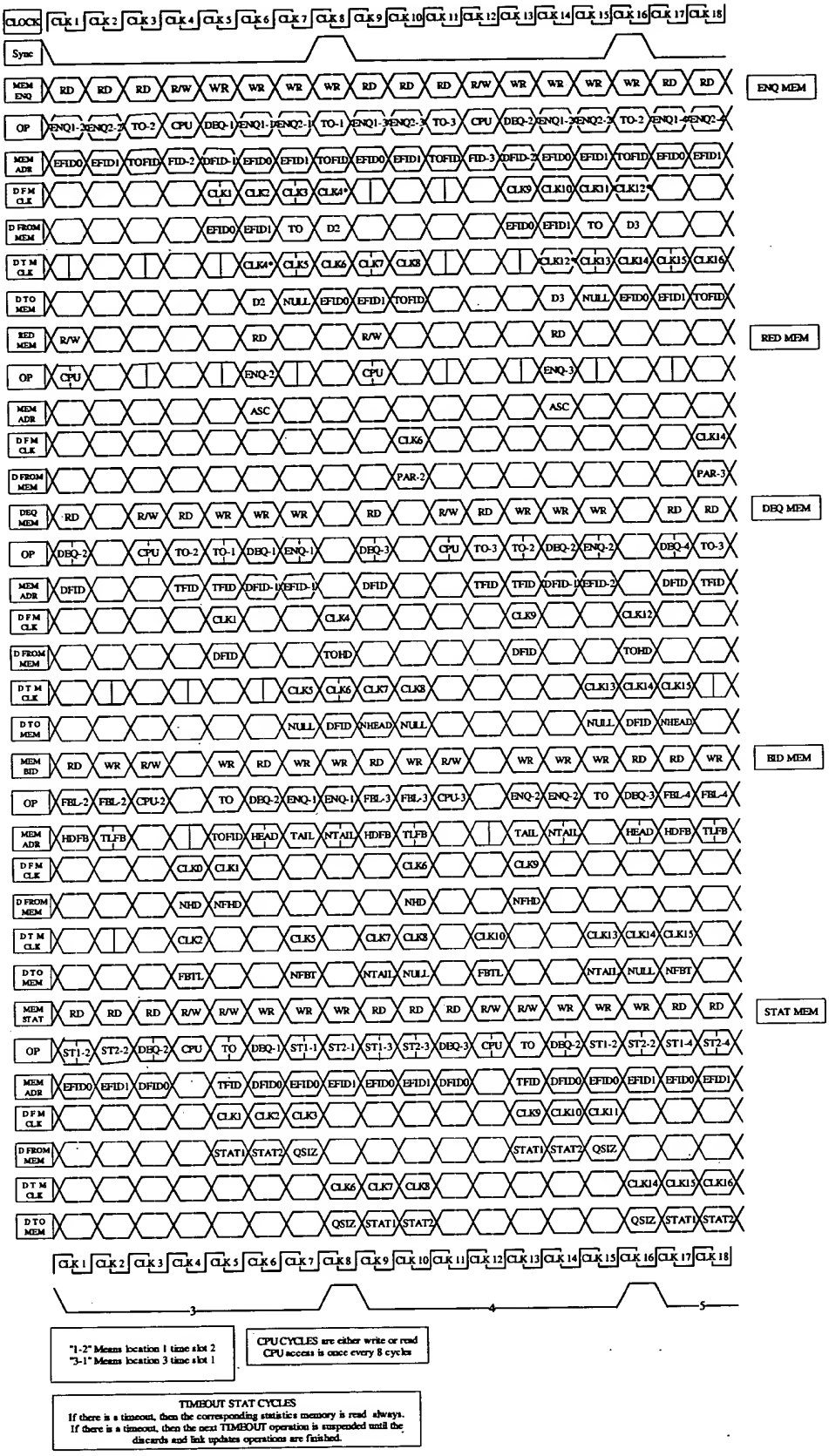

CLOCK
Sync
MEM ENQ
OP
MEM ADR
D F M CLK
D FROM MEM
D T M CLK
D TO MEM
RED MEM
MEM ADR
D F M CLK
D FROM MEM
DEQ MEM
MEM ADR
D F M CLK
D FROM MEM
D T M CLK
D TO MEM
BID MEM
MEM ADR
D F M CLK
D FROM MEM
D T M CLK
D TO MEM
STAT MEM
MEM ADR
D F M CLK
D FROM MEM
D T M CLK
D TO MEM
"1-2" Means location 1 time slot 2
"3-1" Means location 3 time slot 1
CPU CYCLES are either write or read
CPU access is once every 8 cycles
TIMEOUT STAT CYCLES
If there is a timeout, then the corresponding statistics memory is read always.
If there is a timeout, then the next TIMEOUT operation is suspended until the discards and link updates operations are finished.

Figure 341

ENQUEUE								
OP	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
ENQ	RD	RD				WR	WR	
DEQ					WR			
TO			RD					WR
FB								
CPU				R/W				

Figure 342

RED								
OP	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
ENQ						RD		
DEQ								
TO								
FB								
CPU	R/W							

Figure 343

DEQUEUE								
OP	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
ENQ							WR	
DEQ	RD					WR		
TO				RD	WR			
FB								
CPU			R/W					

Figure 344

BID								
OP	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
ENQ							WR	WR
DEQ						RD		
TO					WR			
FB	RD	WR						
CPU			R/W					

Figure 345

STATISTICS								
OP	1	2	3	4	5	6	7	8
ENQ	RD	RD					WR	WR
DEQ			RD			WR		
TO					R/W			
FB								
CPU				R/W				

Figure 346

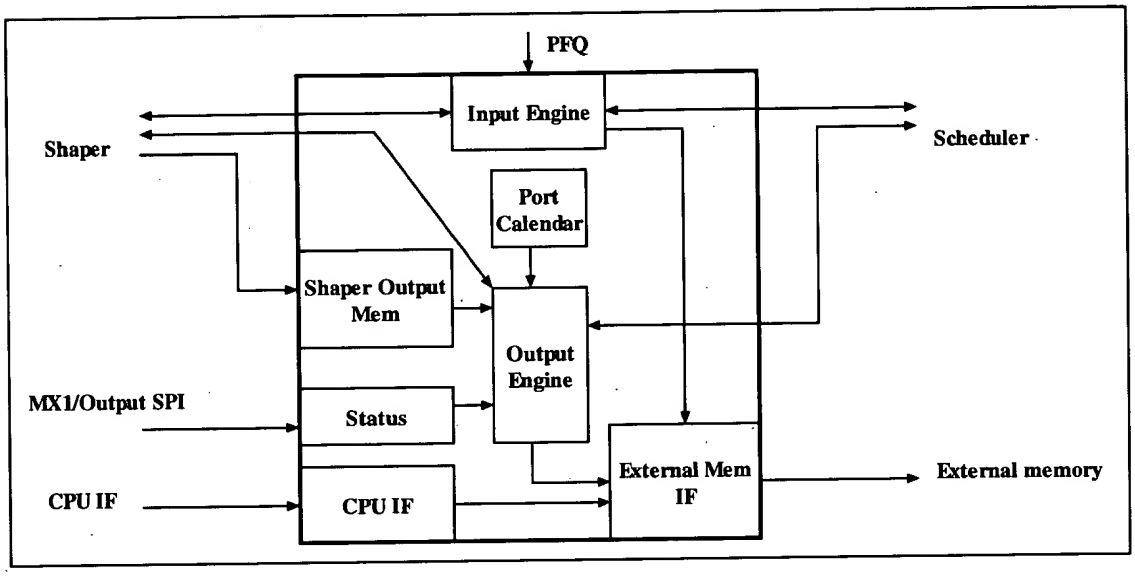


Figure 347

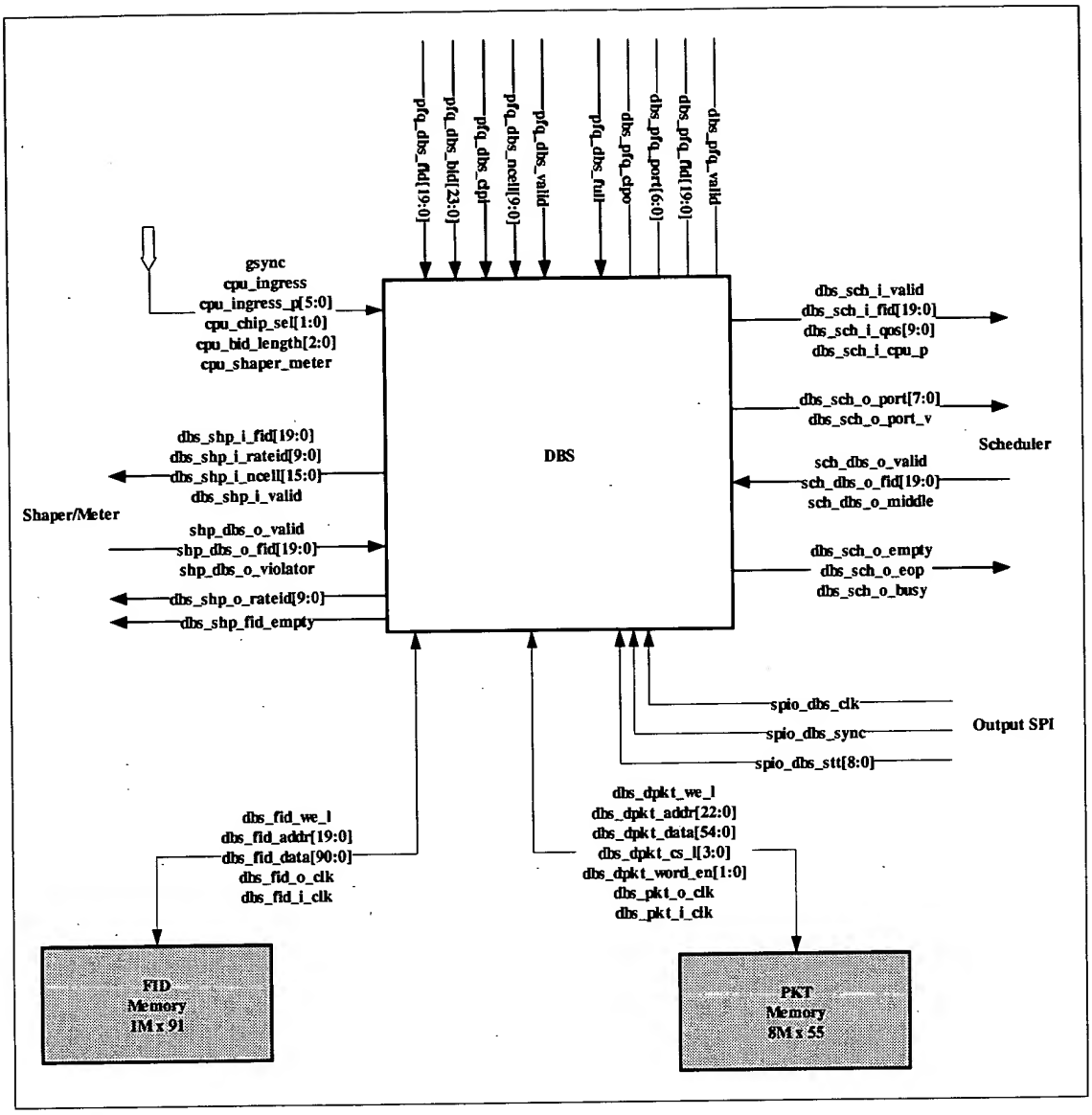

pfq_dbs_pid[19:0]
pfq_dbs_bid[23:0]
pfq_dbs_dpi
pfq_dbs_ncell[19:0]
pfq_dbs_valid
pfq_dbs_full
dbs_pfq_cpo
dbs_pfq_port[6:0]
dbs_pfq_fid[19:0]
dbs_pfq_valid
gsync
cpu_ingress
cpu_ingress_p[5:0]
cpu_chip_sel[1:0]
cpu_bid_length[2:0]
cpu_shaper_meter
dbs_sch_i_valid
dbs_sch_i_fid[19:0]
dbs_sch_i_qos[9:0]
dbs_sch_i_cpu_p
dbs_sch_o_port[7:0]
dbs_sch_o_port_v
Scheduler
dbs_shp_i_fid[19:0]
dbs_shp_i_rateid[9:0]
dbs_shp_i_ncell[15:0]
dbs_shp_i_valid
DBS
sch_dbs_o_valid
sch_dbs_o_fid[19:0]
sch_dbs_o_middle
Shaper/Meter
shp_dbs_o_valid
shp_dbs_o_fid[19:0]
shp_dbs_o_violator
dbs_sch_o_empty
dbs_sch_o_eop
dbs_sch_o_busy
dbs_shp_o_rateid[9:0]
dbs_shp_fid_empty
spio_dbs_clk
spio_dbs_sync
spio_dbs_stt[8:0]
Output SPI
dbs_dpkt_we_l
dbs_dpkt_addr[22:0]
dbs_dpkt_data[54:0]
dbs_dpkt_cs_l[3:0]
dbs_dpkt_word_en[1:0]
dbs_pkt_o_clk
dbs_pkt_i_clk
dbs_fid_we_l
dbs_fid_addr[19:0]
dbs_fid_data[90:0]
dbs_fid_o_clk
dbs_fid_i_clk
FID
Memory
1M x 91
PKT
Memory
8M x 55

Figure 348

Signal Name	#Bits	DIR	Description
General Signals			
rst_l	1	I	Active Low reset for the DBS block
clk	1	I	200 MHz input clock
cpu_ingress	1	I	If set, the block is in ingress chip. If reset, the block is in egress chip.
cpu_ingress_p	6	I	In case of ingress chip, this is an indication of the output port.
cpu_cs_sel	2	I	Control signals to generate chip-select bits to packet memory
cpu_bid_len	3	I	Indication of the amount of supported BIDs
cpu_shp_meter	1	I	If this signal is asserted then traffic has to be shaped depending on shape bit for the fid. Else traffic is just metered.
PFQ			
pfq_dbs_valid	1	I	Signals Valid values on other signals from PFQ to DBS.
pfq_dbs_flowid	20	I	FlowID value to the Shaper block
pfq_dbs_ncell	10	I	Number of cells for the FID
pfq_dbs_bid	23	I	Buffer ID/Pkt ID of the fid.
pfq_dbs_clpi	1	I	Cell Loss Priority bit, which the SHP modifies according to the Dual Leaky Bucket algorithm.
Scheduler			
dbs_sch_i_valid	1	O	Indicates a valid input phase to the scheduler
dbs_sch_i_fid	20	O	FID to schedule
dbs_sch_i_qos	10	O	QOS which the FID belongs to
dbs_sch_i_cpu_p	1	O	If set, the FID belongs to the CPU port, otherwise, use the qos memory to find the proper port.
dbs_sch_o_port	8	O	Current output port, the scheduler should provide a FlowID to that specific port
dbs_sch_o_valid	1	O	If set, Output port valid. If reset, output port is not valid. Scheduler should not perform output phase.
sch_dbs_o_valid	1	I	Output FID from the scheduler is valid
sch_dbs_o_fid	20	I	Output FID from the scheduler
sch_dbs_o_middle	1	I	If set, the FlowID from the scheduler is in the middle of the packet of the last cell of the packet. If reset, this is the first cell of a packet or a cell traffic.
dbs_sch_o_empty	1	O	If set, the FlowID has become empty
dbs_sch_o_eop	1	O	If set, the FlowID has end-of-packet indication

dbs_sch_o_busy	1	O	If set, the port was busy, the scheduler should not continue with he output phase.
Shaper			
dbs_shp_i_valid	1	O	Indicates a valid input phase to the scheduler
dbs_shp_i_fid	20	O	FID to schedule
dbs_shp_i_rateid	10	O	Rate ID which the FID belongs to
dbs_shp_i_ncell	16	O	Number of packets in the cell
shp_dbs_o_valid	1	I	If set, a valid output phase from the shaper is in progress.
shp_dbs_o_fid	20	I	Output FlowID from the shaper
shp_dbs_o_violator	1	I	If set, the output FlowID violated the traffic rate in the shaper
dbs_shp_o_rateid	10	O	RateID for the FlowID of the output phase. In this case the database just reads the memory and feed this field to the shaper directly.
dbs_shp_fid_empty	1	O	If set, the FlowID doesn't have any more packet to shape, the shaper should remove the FlowID from it's links.
External Memory Interfaces			
FID memory			
dbs_fid_addr	20	O	Address
dbs_fid_word_en	2	O	Word enable for write accesses
dbs_fid_we_l	1	O	Write enable
dbs_fid_data	91	IO	data
PKT memory			
dbs_pkt_addr	23	O	Address
dbs_pkt_data	55	IO	data
dbs_pkt_cs_l	3	O	Chip select used when using 2M entries and up
dbs_pkt_word_en	2	O	Word enable for write operation. [0] Controls the write to data bits [23:0] Next PKT [1] Controls the write to data bits [61:24] – ncell, clp, fid
Output SPI block			
spio_dbs_clk	1	I	Input clock from the re-assembly block (a different chip)
spio_dbs_sync	1	I	A sync indication, if set, start of message
spio_dbs_stt	9	I	Status of the output ports in the re-assembly block.

Figure 349

Status cycle	CPU status	Ports status
0	Valid	0-7
1	No status	8-15
2	No status	16-23
3	No status	24-31
4	No status	32-39
5	No status	40-47
6	No status	48-55
7	No status	56-63

Figure 350

Field name	#bits	Owned by	Description
Rp	24	Hardware	Read pointer of the packets link list for the fid
Wp	24	Hardware	Write pointer of the packets link list for the fid
E	1	Hardware	If set, the packet link list is empty. If reset, the packet link list is not empty.
Ncell	10	Hardware	Holds the total amount of cells for the current packet
cell_cnt	10	Hardware	Cells counter, for the shaper or the scheduler. Start decrementing from ncell down to 1
clp	1	Hardware	Clp indication of the current packet, modified in the output phase.
Port	6	Software	An indication of the port that the fid belongs to. Used for the shaper only.
Qos	10	Software	An indication of the QOS that the fid belongs to in case of un-shaped traffic. An indication of the RateID that the fid belongs to in case of shaped traffic.
Cpu_port	1	Software	If set, the fid belongs to the cpu port in the scheduler If reset, the fid belongs to the port specified by the port field.
Shape	1	Software	If set, the fid has to be shaped. If reset, the fid is not to be shaped, it belongs to the scheduler.
Shape_class	3	Software	Priority for the shaped traffic. Each shaped FlowID can exist in one of 8 strict priority classes. A class address is {port, shape_class}

Figure 351

Field name	#bits	Word En[1:0]	Owned by	Description
ncell	10	1	Hardware	Number of cells for the current packet
clp	1	1	Hardware	CLP indication for the packet (or cell)
fid	20	1	Hardware	The fid, which the packet belongs to. Used only for shaped traffic.
Next	24	0	Hardware	Pointer to the next packet

dbs_pkt_word_en	Write to field	Memory bits
[0]	Next	[23:0]
[1]	Ncell, clp, fid	[54:24]

Figure 352

Field name	#bits	Place	Owned by	Description
PortID	6	[5:0]	Software	The value of the output port
valid	1	[6]	Software	If set, the portid is valid. If reset, the portid is not valid
Jump	1	[7]	Software	If set, the address to this memory should be reset in the next positive edge of the clock. If reset, the address to this memory should be incremented by 1 in the next positive edge of the clock.

Figure 353

Field name	#bits	Place	Owned by	Description
Empty	8	[7:0]	Hardware	An indication of empty or not empty per class of the strict priority of the shaped traffic. If a bit is set, the corresponding class is empty.
Prev_qos_v	1	[8]	Hardware	If previous QOS is valid and in middle of packet
Prev_qos_ptr	3	[11:9]	Hardware	This is the QOS to be serviced irrespective of strict priority if prev_qos_v is valid

Figure 354

Field name	#bits	Place	Owned by	Description
p_rp	24	[23:0]	Hardware	A read pointer to the shaped packets link list
p_wr	24	[47:24]	Hardware	A write pointer to the shaped packets link list
p_e	1	[48]	Hardware	If set, the shaped packet link list is empty. If reset, the shaped packet link list is not empty.

Figure 355

Address	Name	Type	Description
0	COM	R/W	[31:28] – Opcode [27:0] – Address, depending on the command. No default value.
1	R0	R/W	General-purpose register. No default value
2	R1	R/W	General-purpose register. No default value
3	R2	R/W	General-purpose register. No default value
4	R3	R/W	General-purpose register. No default value
5 – 31	Reserved		
32	Control	R/W	[0] – Modify CLP enable If reset, no modification is allowed to the CLP bit If set, CLP bit can be modified according to shapers outputs. Default value: 0 [31:1] – Reserved
33	TEST_REG	R/W	[31:0] – Test mode. TBD
34-63	Reserved		

Figure 356

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0001	R		FID[19:0]				
R0	R[7:0]		S Pri	R	Qos[9:0]		A	Port[5:0]
R1	R[7:0]	rp[23:0]						
R2	R[9:8]	wp[23:0]						
R3	R[9:0]			Cell_cnt[9:0]		C	Ncell[9:0]	

Figure 357

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0010	R		FID[19:0]				
R0	R[7:0]	S	Pri	R	Qos[9:0]		A	Port[5:0]
R1	R[7:0]	rp[23:0]						
R2	R[7:0]	wp[23:0]						
R3	R[9:0]			Cell_cnt[9:0]		C	Ncell[9:0]	

Figure 358

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0011	R		FID[19:0]				
R0	R	S	Pri	R	Qos[9:0]		A	Port[5:0]

Figure 359

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0100	R	Pkt[23:0]					
R0	Fid[19:0]					L	Ncell[9:0]	
R1	R	Next[23:0]						

Figure 360

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0101	R	Pkt[23:0]					
R0	Fid[19:0]					L	Ncell[9:0]	
R1	R	Next[23:0]						

Figure 361

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0110	R			Data[7:0]		Port_addr[7:0]	

Figure 362

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0111	R			Data[7:0]		Port_addr[7:0]	

Figure 363

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1000	R		P[3:0]	Empty[7:0]		R	port_add[5:0]

Figure 364

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1001	R		P[3:0]	Empty[7:0]		R	port_add[5:0]

Figure 365

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1010	R					class_addr[8:0]	
R0	R		E	rp[23:0]				
R1	R			wp[23:0]				

Figure 366

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1011	R					class_addr[8:0]	
R0	R		E	rp[23:0]				
R1	R			wp[23:0]				

Figure 367

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1100	R						

Figure 368

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1101	R						

Figure 369

Memory	0	1	2	3	4	5	6	7	0	1	2	3	4	5
FID Memory	R1	R1	*R1*		*W1*		W1	W1						
PKT Memory	*W0*	*W0*	*R1*		R1	R1/	W1	W1n	*W1*	*W1*				
Port Calender								*R3*						
Shp port strict		*R2*			R1		W1	*W1*		*R3*				
Shp out port					R1	*R2*	W1	*W1*						

Figure 370

CLK	DBS	Shaper
6		
7	dbs_shp_i_valid, dbs_shp_I_fid, dbs_shp_i_rateid	Shp_dbs_o_fid, shp_dbs_o_violator, shp_dbs_o_valid
0		
1		
2		
3	dbs_shp_o_rateid	
4	dbs_shp_fid_empty	
5		
6		

Figure 371

CLK	DBS	Scheduler
7	dbs_sch_i_valid, dbs_sch_i_fid, dbs_sch_I_qos, dbs_sch_i_cpu_p	sch_dbs_o_valid, sch_dbs_o_fid, sch_dbs_o_middle
0		
1	dbs_sch_o_port, dbs_sch_o_valid(for next slot)	
2		
3		
4		
5	dbs_sch_o_empty, dbs_sch_eop, dbs_sch_o_busy	
6		

Figure 372

CLK	DBS	PFQ
7	dbs_pfq *valid, dbs*_pfq_fid, dbs_pfq_port, dbs_pfq_clpo,	pfq_dbs_valid, pfq_dbs_ncell, pfq_dbs_clpi, pfq_dbs_bid, pfq_dbs_fid, pfq_dbs_full
0		
1		
2		
3		
4		
5		
6		

Figure 373

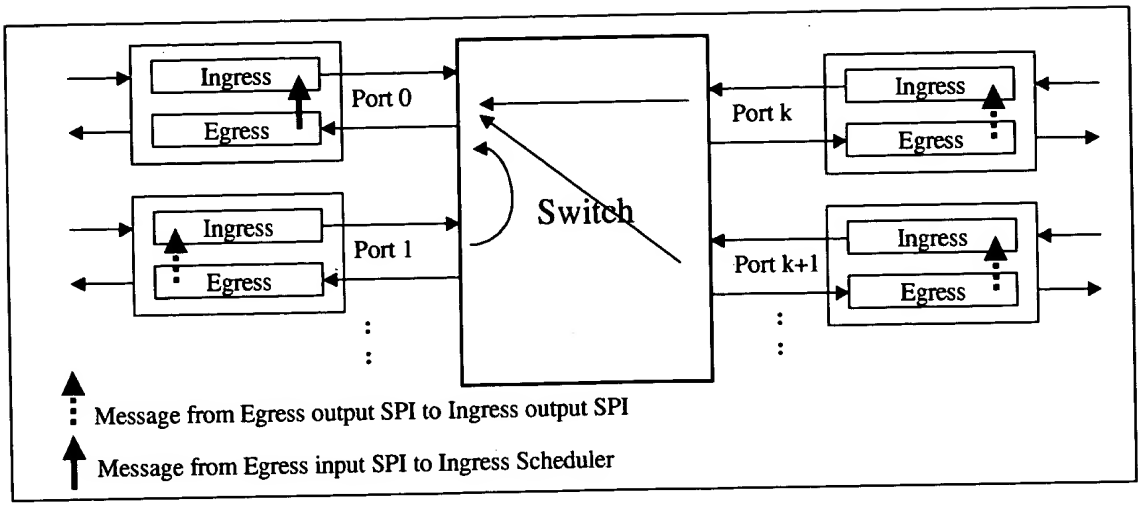


Figure 374

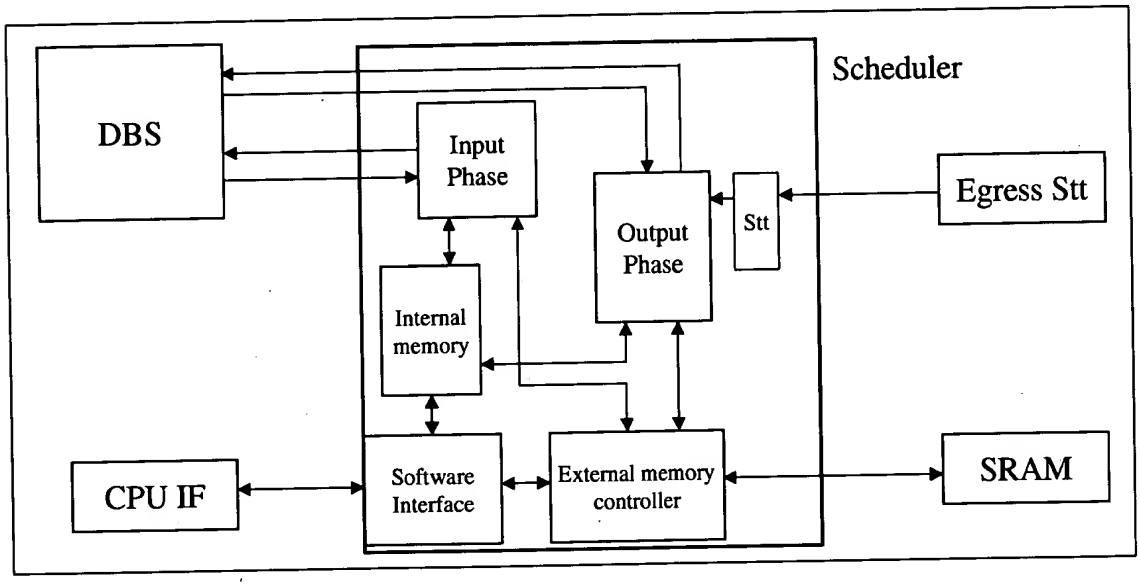


Figure 375

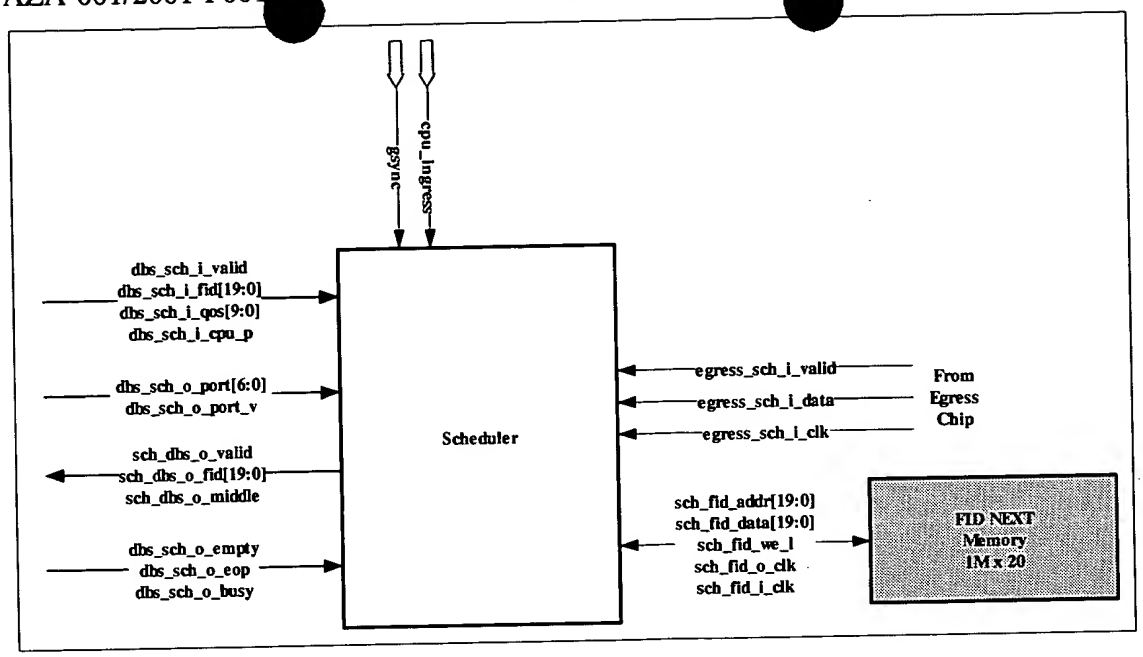


Figure 376

Control (2 pins)			
Signal Name	**#Bits**	**DIR**	**Description**
clk	1	I	200MHz clock
reset	1	I	Asynchronous active high reset signal
DBS			
dbs_sch_i_fid	20	I	FlowID value. This is unshaped flowID which is to be scheduled based QOS in available bandwidth.
dbs_sch_i_qos	10	I	Qos address of the FID.
dbs_sch_i_cpu_port	1	I	If set, the FID belongs to the CPU port and the input/output stages are by-passed.
dbs_sch_i_valid	1	I	If asserted, the fid and all other inputs to the input stage are valid. FlowID is driven by the DBS once every 8 clocks of 200MHz.
dbs_sch_o_port	7	I	This is the port number currently being scheduled by DBS. Scheduler selects flow id based on QOSs strict priority and WRR to schedule from this If bit [6] is set (the MSB) then the Database requests a FlowID for the CPU port.
dbs_sch_o_port_v	1	I	This assertion qualifies dbs_sch_o_port. If this is not asserted then it means DBS is not scheduling any port in this slot.
sch_dbs_o_fid	20	O	The current scheduled FlowID used as a read address for

			the FID memory to get eop, empty and clp info, also to get the next FlowID in the FID linked list.
sch_dbs_o_middle	1	O	This signal is to inform DBS to schedule FID from scheduler though shaper has FID for this port, because scheduler is in middle of packet for this port. This signal will be reset always for the first packet of the port, during which DBS can schedule FID from shaper.
sch_dbs_o_valid	1	O	Valid bit for the read FID address
dbs_sch_o_eop	1	I	The end of packet indication for the scheduled FlowID.
dbs_sch_o_empty	1	I	The empty packet indication of the scheduled FlowID.
dbs_sch_o_busy	1	I	This is qualifying signal for all writes in the scheduler in the current slot. If this signal is asserted it means DBS is scheduling shaped traffic and scheduler not to update any parameters.
Egress Status			
egress_sch_i_clk	1	I	Input clock for the status bits
egress_sch_i_data	1	I	Input data for status
egress_sch_i_valid	1	I	When set input data is valid
CPU Interface (73 pins)			
cpu_addr	6	I	Address for CPU commands, valid when cs_l is asserted.
cpu_data_in	32	I	Data in for write commands, valid when cs_l is asserted.
cpu_sch_data_out	32	O	Data out for read commands, valid when cs_l is asserted.
cpu_sch_cs_l	1	I	Active low chip select, when asserted all other signals are valid
cpu_rdwr_l	1	I	If set, the CPU issued a read command, if reset the CPU issued a write command, valid when cs_l is asserted.
gsync	1	I	One cycle pulse asserted once every 520 cycles of 200 MHz. It has to be sampled before use.
mode	1	I	If set, indicate the chip is in ingress mode.
External Memory Interface (43 chip pins)			
sch_fid_addr	20	O	Address to the FID next memory.
sch_fid_data_in	20	I	Read data in from the memory.
sch_fid_data_out	20	O	Write data to the memory.
sch_fid_we_l	1	O	Write enable: 0 – write and 1 – read.
sch_fid_data_en	1	O	Data write enable for the bi-dir pads.
sch_fid_o_clk	1	O	The clock to the memory (200 MHz) - External
sch_fid_i_clk	1	O	The clock to the memory (200 MHz) - Internal

Figure 377

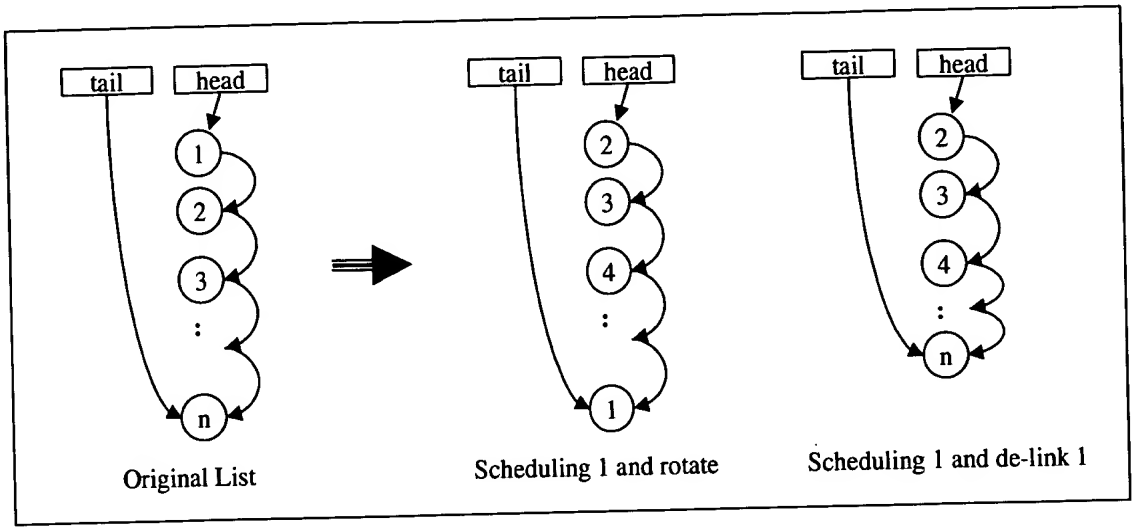


Figure 378

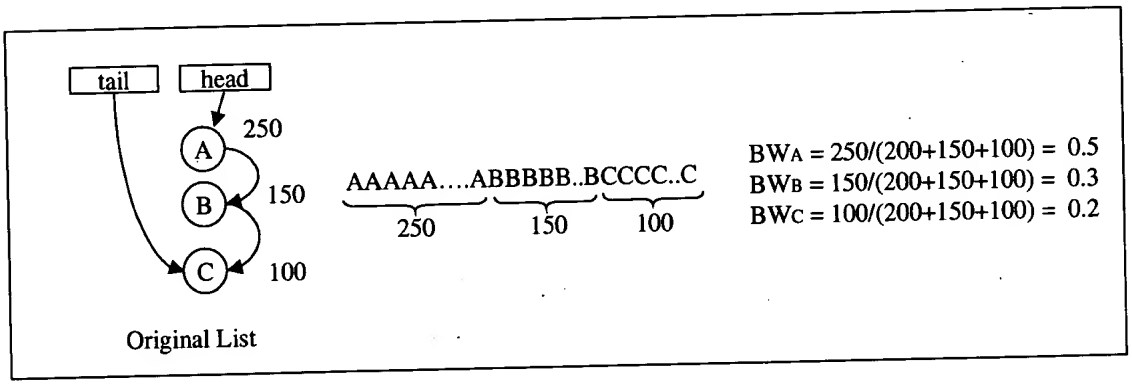


Figure 379

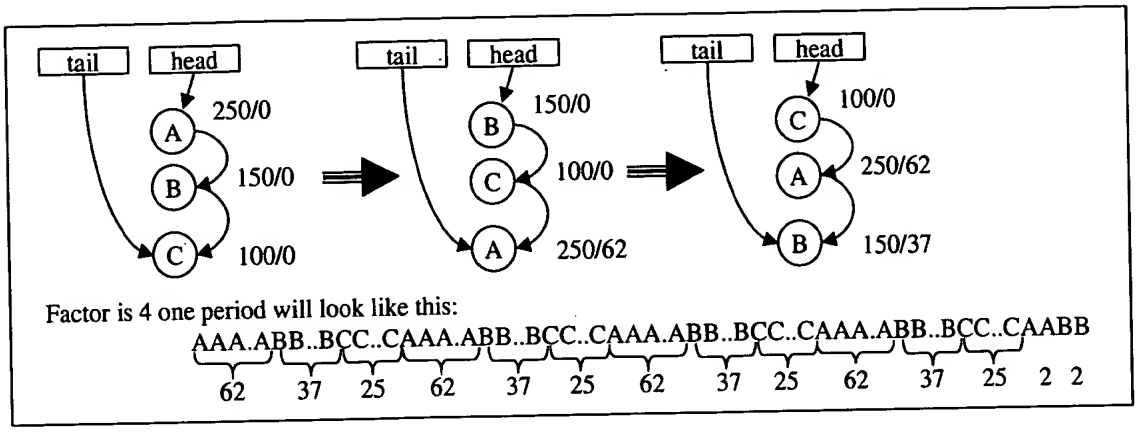


Figure 380

#	Description
1	No Strict priority WRR QoS0-QoS6 QOS7 is best effort
2	Strict priority: QoS0, WRR QoS1-QoS6, QOS7 is best effort
3	Strict priority: QoS0-QoS1, WRR QoS2-QoS6, QOS7 is best effort
4	Strict priority: QoS0-QoS2, WRR QoS3-QoS6, QOS7 is best effort
5	Strict priority: QoS0-QoS3, WRR QoS4-QoS6, QOS7 is best effort
6	Strict priority: QoS0-QoS4, WRR QoS5-QoS6, QOS7 is best effort
7	Strict priority: QoS0-QoS5, WRR QoS6, QOS7 is best effort This is exactly like a complete strict priority for all QOSs.

Figure 381

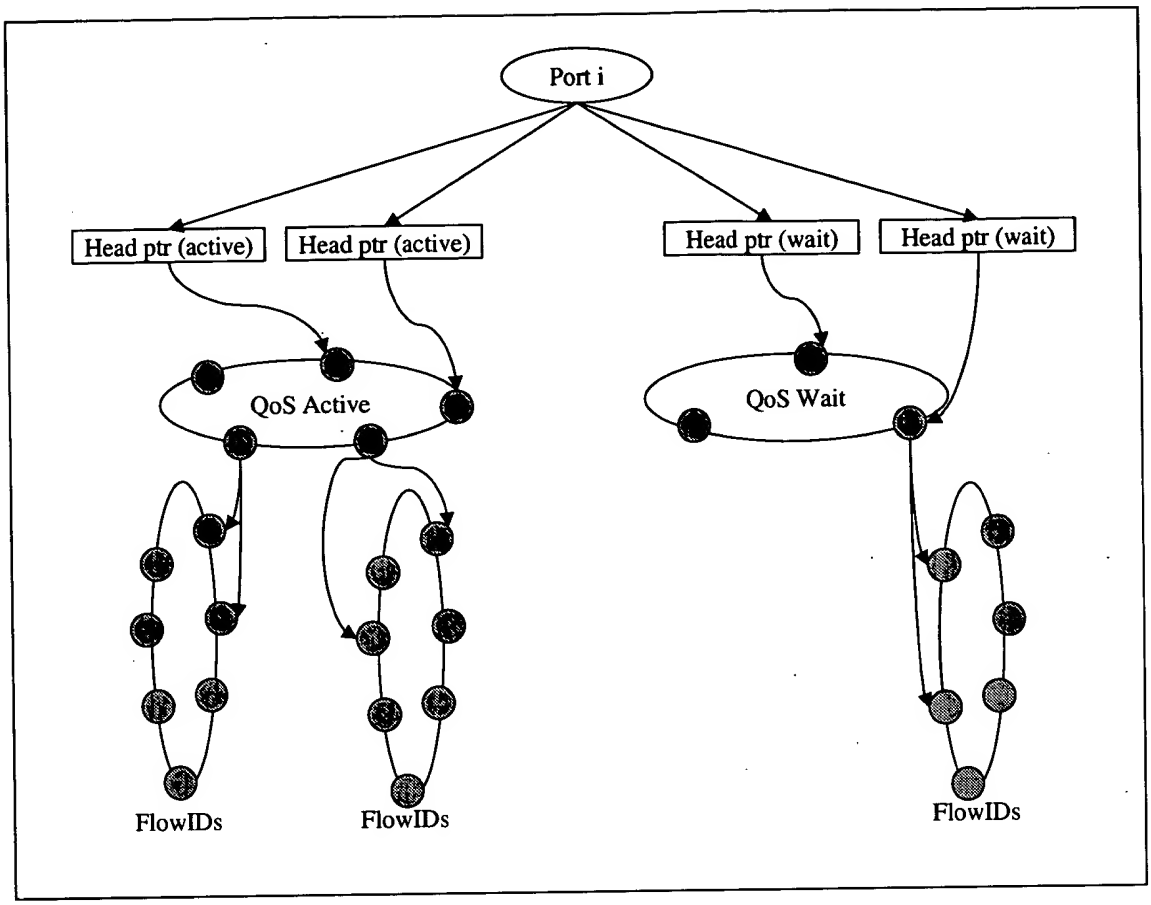

Port i
Head ptr (active)
Head ptr (active)
Head ptr (wait)
Head ptr (wait)
QoS Active
QoS Wait
FlowIDs
FlowIDs
FlowIDs

Figure 382

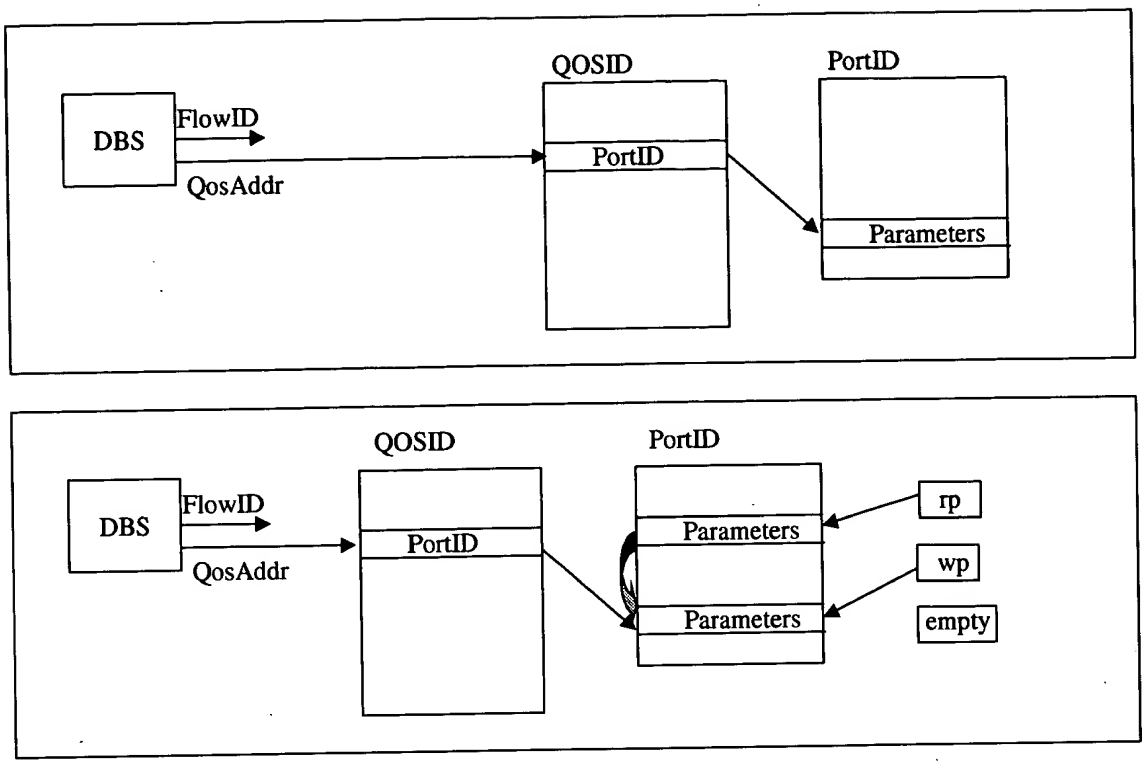


Figure 383

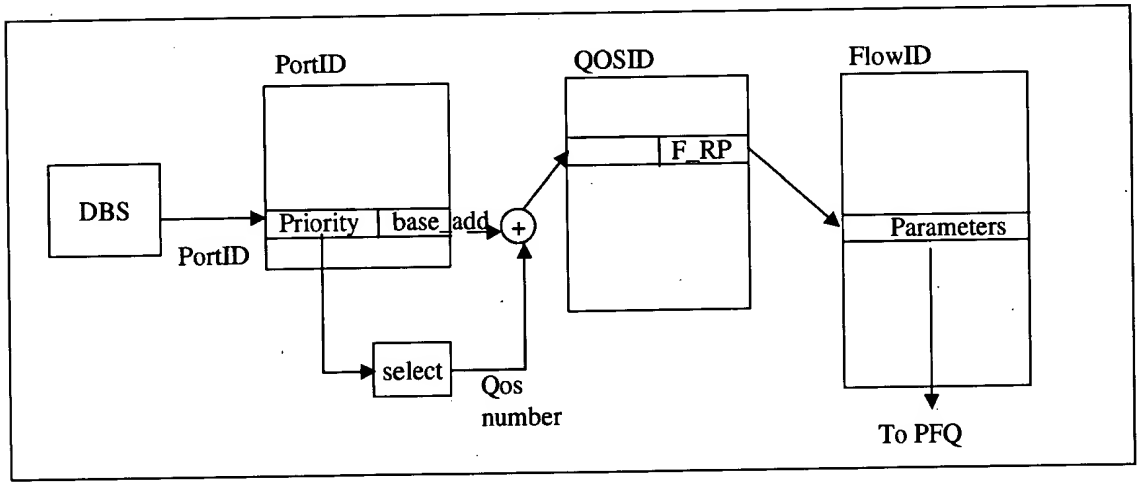


Figure 384

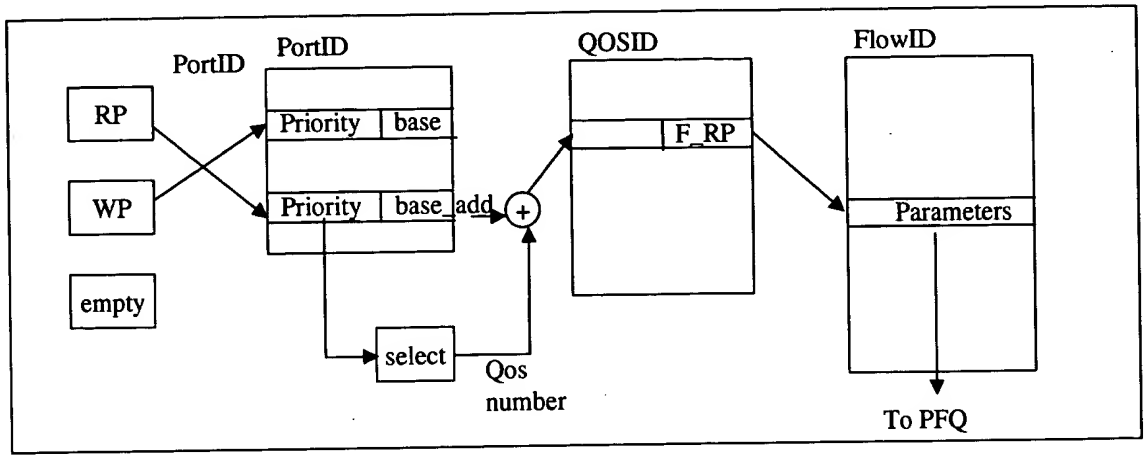


Figure 385

Field name	#bits	Range	Owned by	Description
F_NEXT	20	19:0	Scheduler	The next Flow ID, which is linked to the previous Flow ID. The address to the memory is the current Flow ID.

Figure 386

Field name	#bits	Range	Owned by	Description
PortID	6	16:11	Software	Indication of which port the specific QOS is assigned to. Since total number of ports is 256 (64*4) there are 8 bits descriptor.
Q_WEIGHT	8	10:3	Software	Weight of the QOS for the weighted round robin algorithm.
Q_NUM	3	2:0	Software	QOS number in the sequence of the QOSs per port. Since it's only 3 bits, the limit is 8 QOSs per port.

Figure 387

Field name	#bits	Range	Owned by	Description
Q_WEIGHT_M	25	65:41	Scheduler	A count down counter to keep the amount of credits left for the QOS. Since this field can become negative there is an extra bit to save the sign.
F_RP	20	40:21	Scheduler	The head of the Flow ID link list for the QOS.
F_WP	20	20:1	Scheduler	The tail of the Flow ID link list for the QOS.
F_EMPTY	1	0	Scheduler	An indication of an empty link list.

Figure 388

Field name	#bits	Range	Owned by	Description
PRIORITY	8	20:13	Software	Selecting Strict priority and WRR to the QoSs, see table above for complete description of his field.
FACTOR	3	12:10	Software	In order to make the traffic less bursty, each QOS weight will be divided by 2^FACTOR. That means that there will be some rotation of QOSs in the active link list before moving them to the waiting list.
Q_BASE_ADDR	10	9:0	Software	A pointer to the QOS memory, which is the first QOS that is assigned to this port.

Figure 389

Field name	#bits	Range	Owned by	Description
PREV_QOS	3	46:44	Scheduler	The QOS that was services last to be serviced for this port
PREV_QOS_V	1	43	Scheduler	If set, PREV_QOS has to be scheduled with no connection to priority since in the middle of a packet. If reset, choose a QOS according to priority and WRR.
ACTIVE_PTR	3	42:40	Scheduler	This pointer to points to current QOS to be served in round robin arbitration. When Q_WEIGHT_M is either zero or negative and eop is received for the current flow ID, this pointer moves to next QOS in round robin. This pointer also moves when empty (no more packets pending for this flow ID) is received from DBS and F_RP & F_WP are same, irrespective of q_weight_m as link list becomes empty for this QOS. Note that this

				pointer moves around only on round robin QOSs. When strict priority QOS is served, this pointer is not altered.
Q_WEIGHT_MF	24	39:16	Scheduler	A count down counter of weight/factor for the current QOS. When this count becomes zero and eop arrives, next qos available will be serviced. And new QOS Weight/port factor is loaded to this counter.
QA_EMPTY	8	15:8	Scheduler	Indication per QOS if it's empty or some flow ids are linked to it. (Used for the priority scheme). This is active empty list and when any of the QOS weight becomes zero or negative, it is moved to QW_EMPTY list. QW_EMPTY is loaded to this list when all round robin QOS's weight become zero or negative. If any strict priority QOS is pending in QW_EMPTY list it is serviced first before resuming round robin operation.
QW_EMPTY	8	7:0	Scheduler	This is waiting empty list. Scheduler Input phase updates this list when new QOS arrives. Also non_empty QOSs are moved from active list to this list when there is eop and weight is decremented to zero or negative.

Figure 390

Field name	#bits	Range	Owned by	Description
P_NEXT	6	5:0	Scheduler	Used for port calendar. In case of an Ingress Chip, the ports are linked together to a virtual ports link list. An input phase to an empty port will cause the linking of the port, an output phase with cell_cnt = 0 will de-link the virtual port from the list.

Figure 391

Address	Name	Type	Description
0	COM	R/W	[31:28] – Opcode [27:0] – Address, depending on the command. Default value for bits [31:28] – 0 No default value for bits [27:0]
1	R0	R/W	General-purpose register. No default value
2	R1	R/W	General-purpose register. No default value

3	R2	R/W	General-purpose register. No default value
4 – 31	Reserved		
32	CONTROL	R/W	[0] – OUT_EN Global output enable for all ports, if reset, no output stage will be performed for all ports Default value 0 (output disabled). [31:1] – Reserved
33	INGRESS_PTR	R/W	[7:0] – Head pointer of the virtual port list. Used only in case of an ingress chip. No default value. [15:8] – Tail pointer of the virtual port list. Used only in case of an ingress chip. No default value. [16] – Empty indication of the virtual port list. Used only in case of an ingress chip. If set, the virtual port list is empty. Default value 1 (empty list) [31:17] – Reserved.
34	CPU_R_PTR	R/W	[19:0] – Head pointer to the FlowID list of the CPU port. No default value.
35	CPU_W_PTR	R/W	[19:0] – Tail pointer to the FlowID list of the CPU port. No default value.
36	CPU_CTRL	R/W	[0] – Empty indication of the CPU FlowID list. If set, the FlowID linked list is empty. Default value 1 (empty list) [1] – CPU output port enable. If set, the scheduler can schedule FlowIDs for the CPU port. Default value 0 (Disable the CPU port) [31:2] - Reserved
37	WEIGHT_QUOTA	R/W	[15:0] – weight_quota This value is a multiplicand to calculate the weight per QOS. Default value 1 [31:16] - Reserved
38	TEST_REG	R/W	[31:0] – Test mode. TBD

39 - 63	Reserved		

Figure 392

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0001	R		Fid[19:0]				
R0	R			Next[19:0]				

Figure 393

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0010	R		Fid[19:0]				
R0	R			Next[19:0]				

Figure 394

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0010	R		Fid[19:0]				
R0	R			Next[19:0]				

Figure 395

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0100	R[17:0]				Addr[9:0]		
R0	R[12:0]			qnum	Weight		R[1 :0]	PortID

Figure 396

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0101	R[17:0]				Addr[9:0]		
R0	R[10:0]			E	Read PTR[19:0]			
R1	R[11:0]			Write PTR[19:0]				
R2	R[6:0]	W_M[24:0]						

Figure 397

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0110	R[17:0]				Addr[9:0]		
R0	R[10:0]			E	Read PTR[19:0]			
R1	R[11:0]			Write PTR[19:0]				
R2	R[6:0]	Q_WEIGHT_FM[24:0]						

Figure 398

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0111	R[21:0]					Addr[5:0]	
R0	Priority[7:0]		R[4:0]	Factor[2:0]	R[5:0]		Q_B_Addr[9:0]	

Figure 399

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1000	R[21:0]					Addr[5:0]	
R0	Priority[7:0]		R[4:0]	Factor[2:0]	R[5:0]		Q_B_Addr[9:0]	

Figure 400

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1001	R[21:0]					Addr[5:0]	
R0	R[8:0]			PQ[6:0]		Qa_Empty[7:0]	Qw_Empty[7:0]	
R1	R[7:0]	Q_WEIGHT_FM[23:0]						

Figure 401

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1010	R[21:0]					Addr[5:0]	
R0	R[8:0]			PQ[6:0]		Qa_Empty[7:0]	Qw_Empty[7:0]	
R1	R[7:0]	Q_WEIGHT_FM[23:0]						

Figure 402

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1011	R[21:0]					Addr[5:0]	
R0	R[25:0]						Next[5:0]	

Figure 403

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1100	R[21:0]					Addr[5:0]	
R0	R[25:0]						Next[5:0]	

Figure 404

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0100	R[17:0]					Addr[9:0]	
R0	R[12:0]			qnum	Weight[7:0]		PortID[7:0]	

Figure 405

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1101	R[21:0]					Addr[5:0]	
R0	R[5:0]	Q_B_Addr[9:0]			Priority[7:0]		Factor[7:0]	

Figure 406

Memory	CLK0	CLK1	CLK2	CLK3	CLK4	CLK5	CLK6	CLK7
External Memory (The FID parameter memory is in the database block)								
FID Next	*Wr*	*Rd*		CPU Rd/Wr			Wr	
Internal Memories								
QOS Parameter	CPU Rd/Wr	Rd				*Rd*		
QOS Descriptor	CPU Rd/Wr	Rd				*Rd*	Wr	*Wr*
Port Descriptor	CPU Rd/Wr	*Rd*		Rd			Wr	*Wr*
Port Next	CPU Rd/Wr	*Rd*					Wr	*Wr*
Rd/Wr – input stage *Rd/Wr* – output state								
Input Phase Operations								
FID Next memory	CPU Rd/Wr						Wr [f_wp] {Next}	
QOS Desc memory	CPU Rd/Wr	Rd [Qos] {descr}		Register [Qos] {descr}			Wr [Qos] {descr}	
Port memory	CPU Rd/Wr			Rd [Port] {descr}		Register [Port] {descr}	Wr [Port] {descr}	
Output Phase Operations								
FID Next memory	*Wr-1 [f_wp] {Next}*	*Rd0 [f_rp] {Next}*		CPU Rd/Wr				
QOS Desc memory	CPU Rd/Wr					*Rd1 [Qos] {descr}*		*Wr0 [Qos] {descr}*

Port memory	CPU Rd/Wr	*Rd1 [Port] {descr}*						*Wr0 [Port] {descr}*

Figure 407

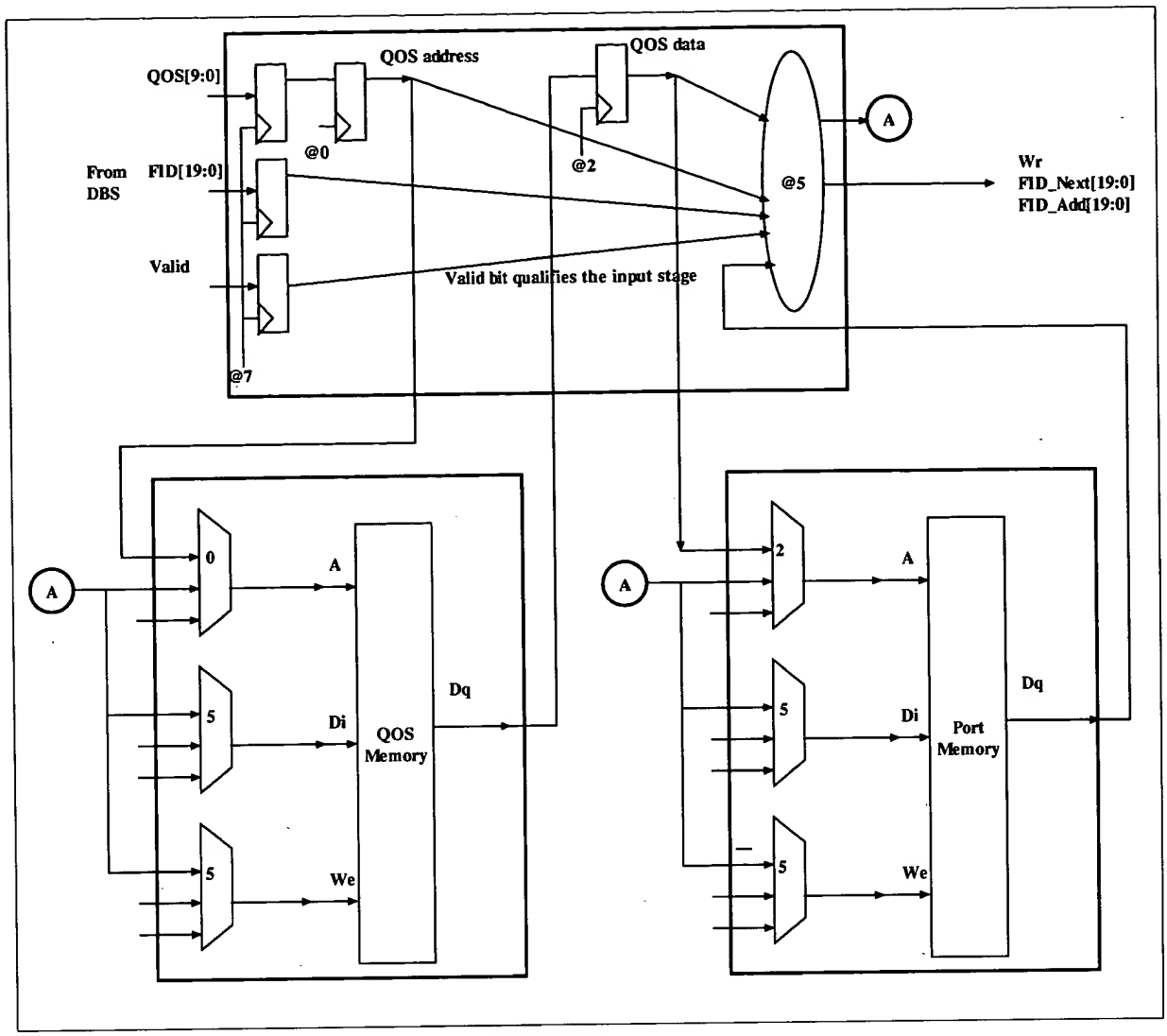

QOS[9:0]
QOS address
QOS data
@0
@2
@5
@7
From DBS
FID[19:0]
Valid
Valid bit qualifies the input stage
Wr
FID_Next[19:0]
FID_Add[19:0]
A
0
5
5
Di
We
Dq
QOS Memory
2
Port Memory

Figure 408

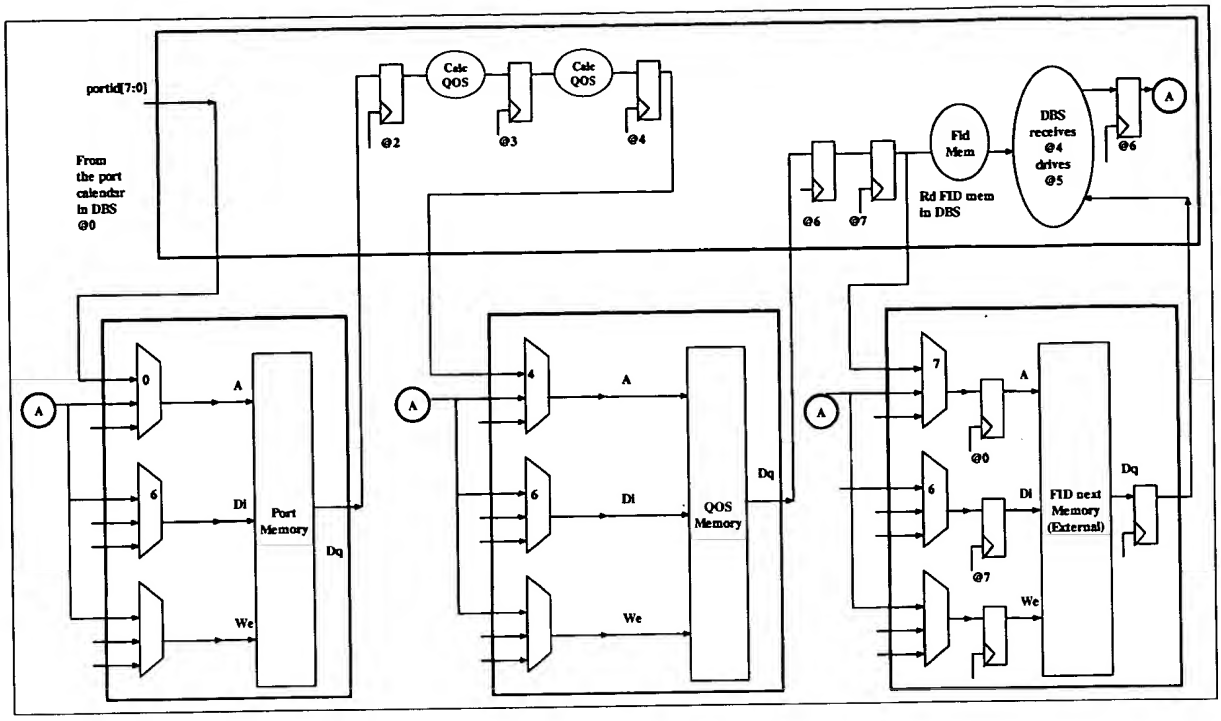


Figure 409

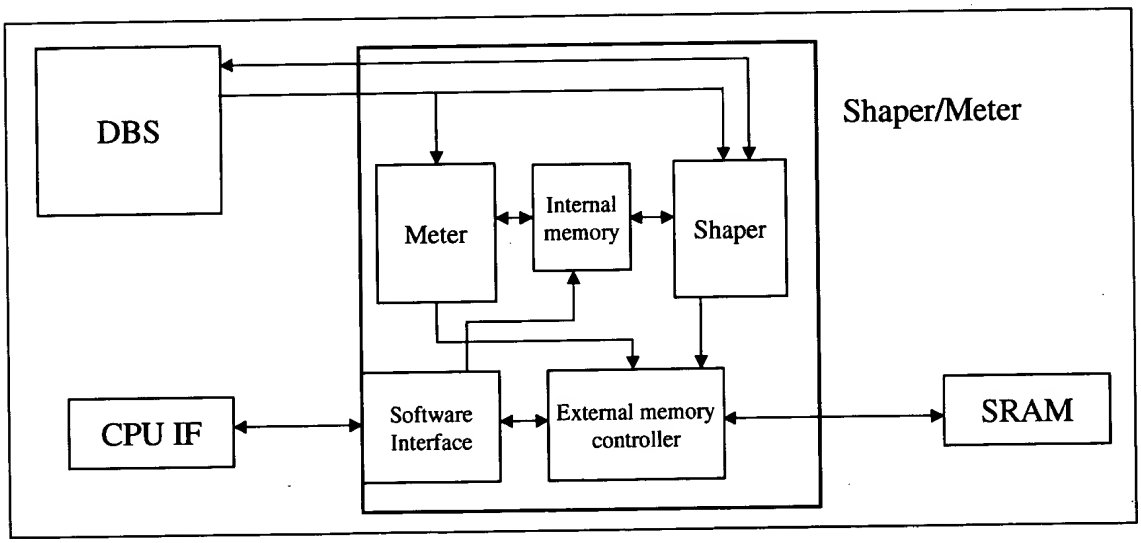


Figure 410

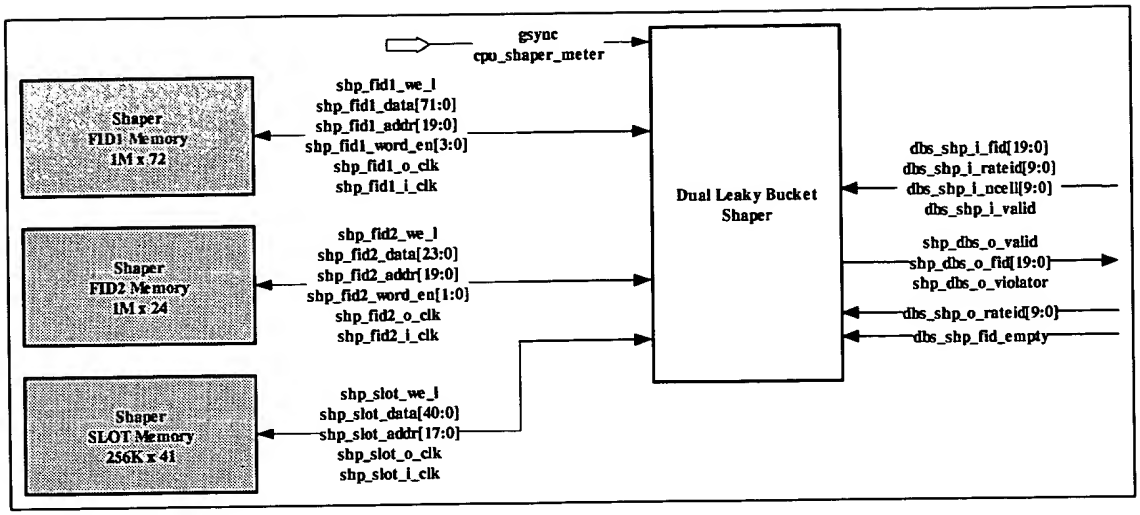


Figure 411

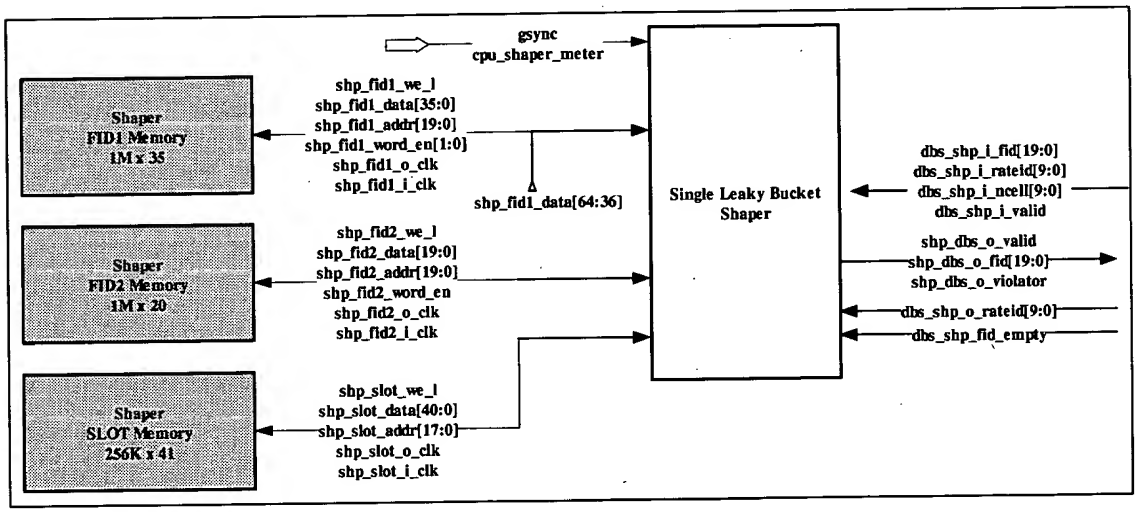


Figure 412

General Signals			
Signal Name	#Bits	DIR	Description
rst_l	1	I	Active Low reset for the Shaper block
clk	1	I	200 MHz input clock
gsync	1	I	Global sync signal
shp_dbs_shp_meter	1	I	A control signal to DBS. If 1, this block acts as a shaper If 0, this block acts as a meter
DataBase (DBS) – Input Phase Signals			
dbs_shp_i_valid	1	I	Signals Valid values on other Input Phase signals from DBS to Shaper, otherwise ignore other signals.
dbs_shp_i_fid	20	I	Flow ID for the Input Phase
dbs_shp_rateid	10	I	RateID for the Flow ID being sent along used for output phase also
dbs_shp_i_ncell	16	I	Number of cells for the current input FlowID
DataBase (DBS) - Output Phase Signals			
shp_dbs_o_valid	1	O	If set, validates the output phase of the shaper, the FlowID and the violator signals are valid.
shp_dbs_o_fid	20	O	Flow ID for the Output Phase
dbs_shp_o_fid_empty	1	I	Signal from the Database indicating no more cells available for this fid to shape. The RateId comes along with this indication.
dbs_shp_o_rateid	10	I	RateID for the output FlowID. Since the RateID field is kept in the database memory, the database block needs to read the FlowID memory and drive that field.
shp_dbs_o_violator	1	O	An indication that the current FlowID is a violator of the rate assigned to it. This will cause the database block to mark the CLP bit for that FlowID.
Shaper Slot Memory			
shp_slot_data	41	I/O	Data to the Shaper Slot Memory
shp_slot_addr	18	O	Address Bus
shp_slot_we_l	1	O	Active low write enable
shp_slot_i_clk	1	O	200 MHz input clock to the Shaper memories
shp_slot_o_clk	1	I	200 MHz output clock from the Shaper memories. This routed back clock is used to adjust the PLL.
Shaper FID1 Memory			
shp_fid1_data	72	O	Data bus. Bits [71:36] are optional for dual

			leaky bucket algorithm
shp_fid1_addr	20	O	Address Bus
shp_fid1_we_l	1	O	Active low write enable
shp_fid1_word_en	4	O	Word enable for write operation. [0] Controls the write to data bits [17:0] [1] Controls the write to data bits [35:18] [2] Controls the write to data bits [53:36] [3] Controls the write to data bits [63:54]
shp_fid1_i_clk	1	O	200 MHz input clock to the Shaper memories
shp_fid1_o_clk	1	I	200 MHz output clock from the Shaper memories. This routed back clock is used to adjust the PLL.
Shaper FID2 Memory			
shp_fid2_data	24	O	Data Bus
shp_fid2_addr	20	O	Address Bus
shp_fid2_we_l	1	O	Active low write enable
shp_fid2_word_en	2	O	Word enable for write operation. [0] Controls the write to data bits [22:0] [1] Controls the write to data bits [23] which is the start bit
shp_fid2_i_clk	1	O	200 MHz input clock to the Shaper memories
shp_fid2_o_clk	1	I	200 MHz output clock from the Shaper memories. This routed back clock is used to adjust the PLL.
CPU Block Interface			
cpu_shp_cs_l	1	I	Active Low Chip Select from the CPU for the SHP block.
cpu_rdwr_l	1	I	Read/Active Low Write signals from the CPU
cpu_addr[5:0]	6	I	Address of the Internal Shaper Register for CPU Access
cpu_data_in[31:0]	32	I	Input Data from the CPU to the Shaper block
cpu_shp_data_out[31:0]	32	O	Output Data from the Shaper block to the CPU

Figure 413

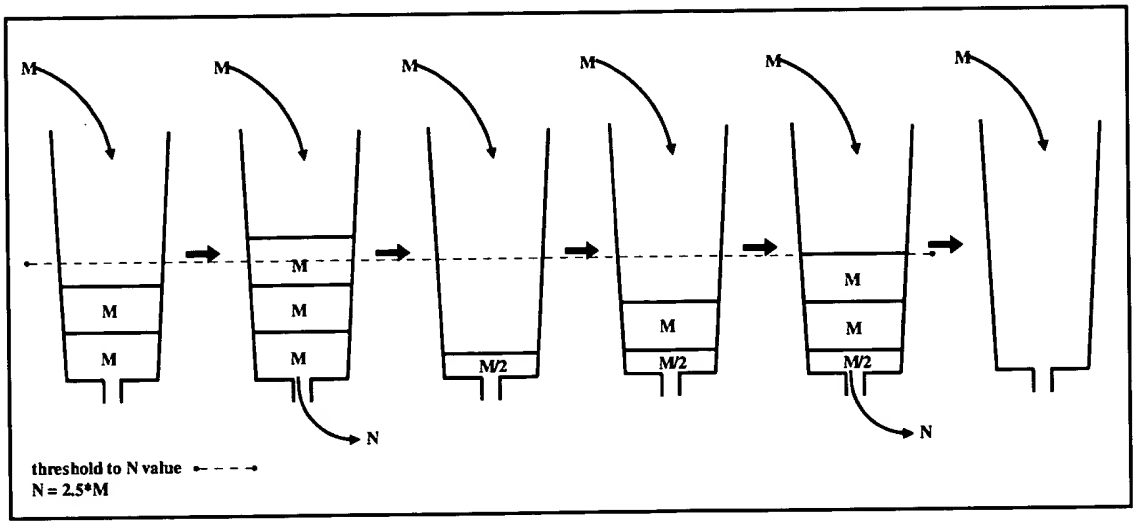


Figure 414

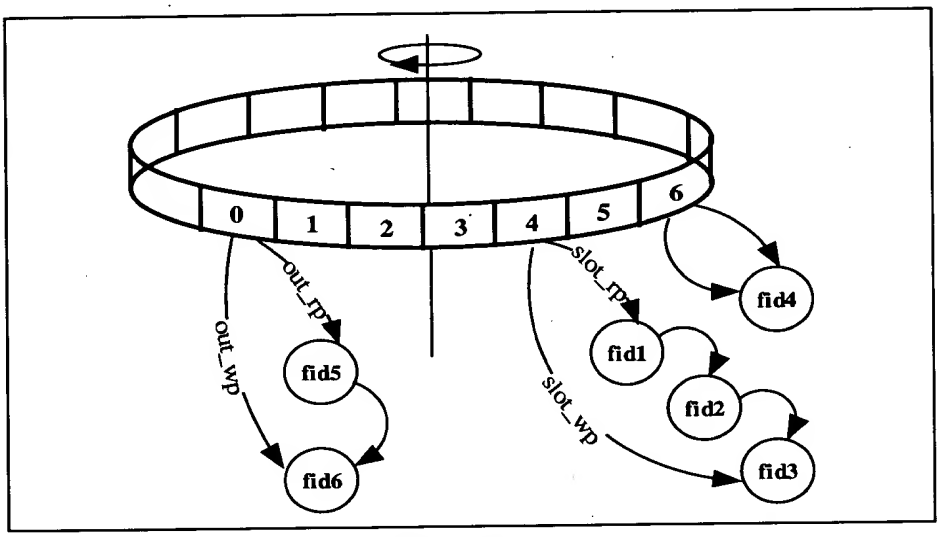


Figure 415

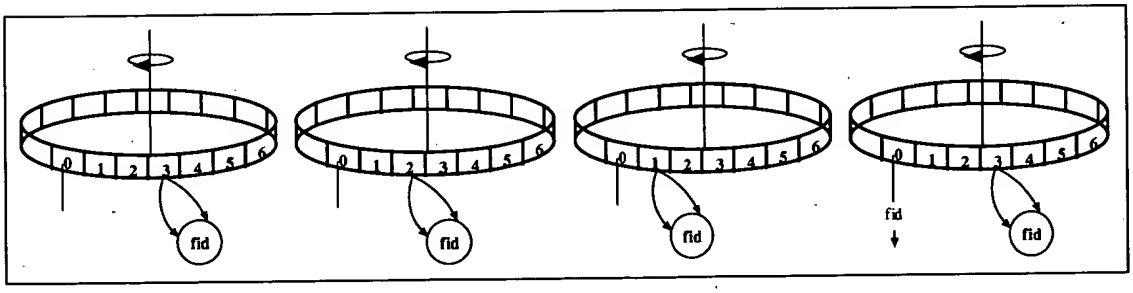


Figure 416

Time Slot	0	1	2	3	4	5	6	7	8	9	10
Future time	1.28	1.56	1.84	2.12		1.40	1.68	2.96		2.24	
Residue	0.28	0.56	0.84		0.12	.40	0.68		0.96		0.24
FlowID out	V	V	V		V	V	V		V		V
	V – valid output										

Figure 417

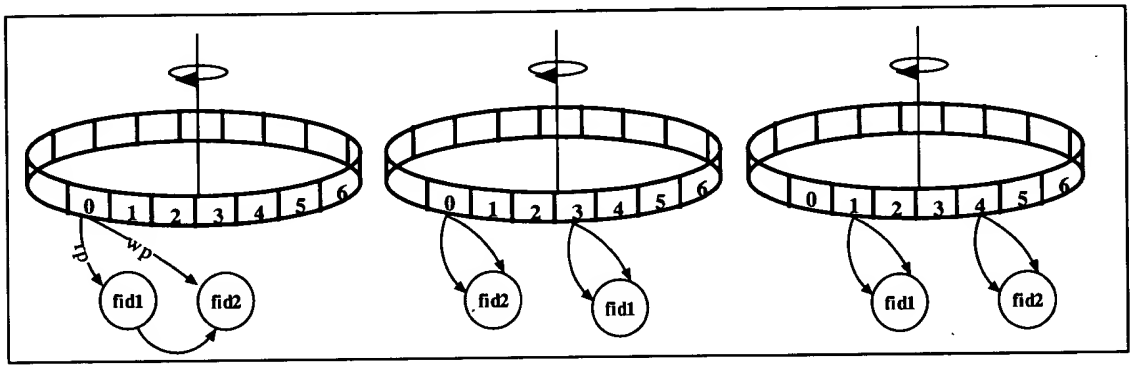


Figure 418

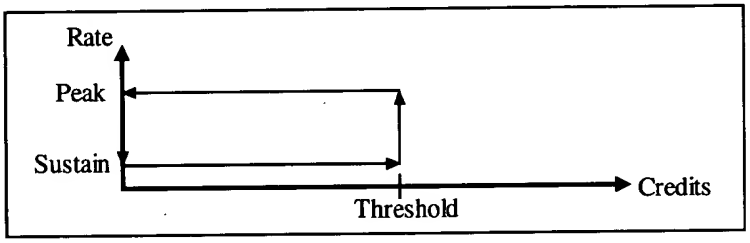


Figure 419

Rate	STS1 51.67M	OC3 155M	OC12 620M	OC48 2.480G	OC192 9.92G	T1 2.048M	10M Ethernet
K	247.74	82.58	20.65	5.16	1.290	6250	128

Figure 420

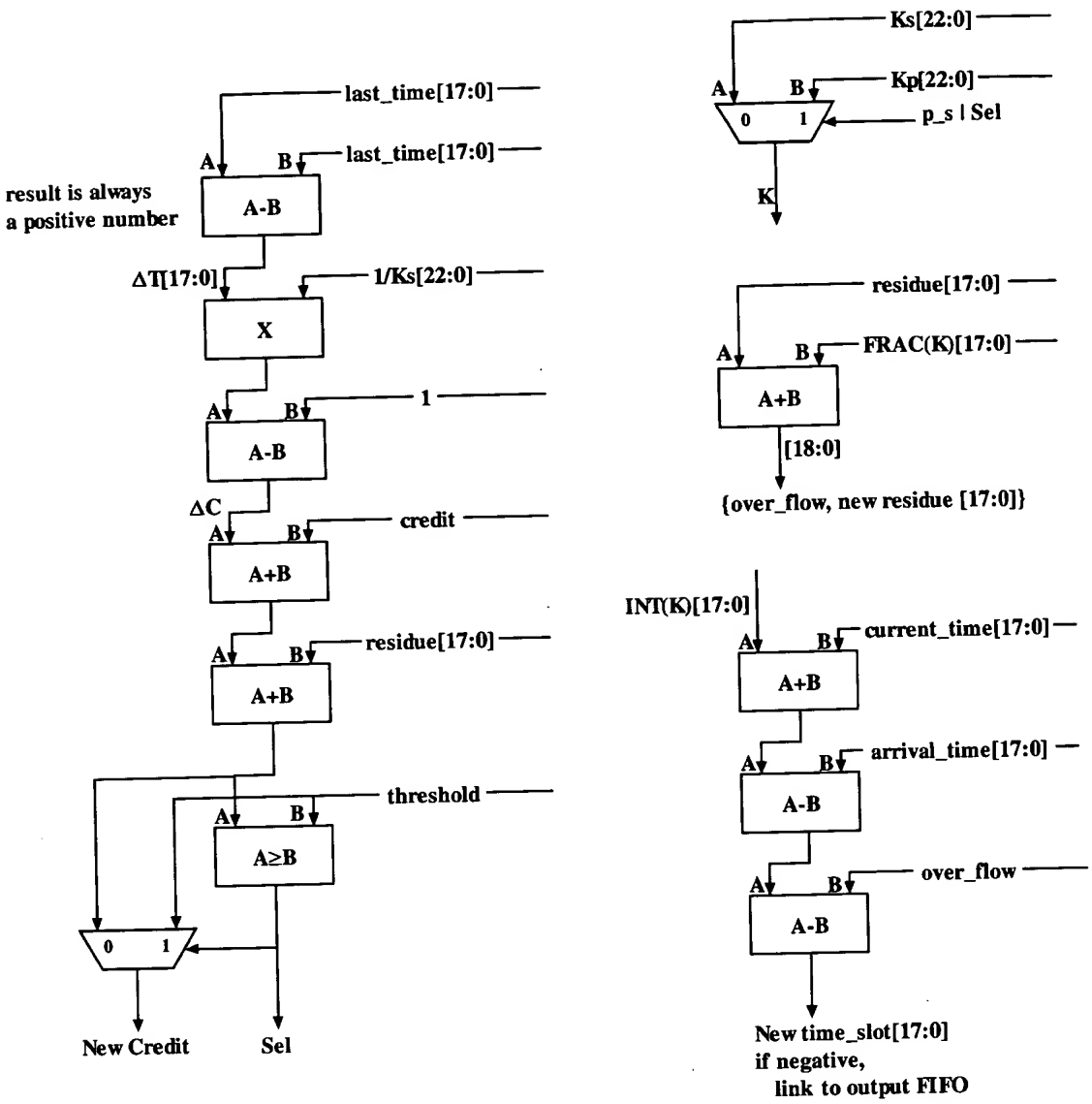

last_time[17:0]
last_time[17:0]
A
B
result is always
a positive number
A-B
ΔT[17:0]
1/Ks[22:0]
X
1
A-B
ΔC
credit
A+B
residue[17:0]
A+B
threshold
A≥B
0
1
New Credit
Sel
Ks[22:0]
Kp[22:0]
p_s | Sel
K
residue[17:0]
FRAC(K)[17:0]
A+B
[18:0]
{over_flow, new residue [17:0]}
INT(K)[17:0]
current_time[17:0]
A+B
arrival_time[17:0]
A-B
over_flow
A-B
New time_slot[17:0]
if negative,
link to output FIFO

Figure 421

BG_CNT	Interval of time [mSec]
0	40
1	80
2	120
3	160
4	200
5	240
6	280
7	320
8	360
9	400
10	440
11	480
12	520
13	560
14	600
15	640

Figure 422

Mode	External Memory Usage			Internal Memory
	Slot memory 41 bits	FID1 memory 65 bits	FID2 memory 24 bits	RateID/ Threshold
Shaper – Single leaky bucket	41 bits of data	36 bits of data only	21 bits of data only	23 bits used only
Shaper – Dual leaky bucket	41 bits of data	65 bits of data	24 bits of data	79 bits used
Meter	Not in use	36 bits of data only	20 bits of data only	27 bits used

Figure 423

Field name	#bits	Owned by	Description
ExpKs	5	Software	Exponent of Sustain rate increment
ManKs	18	Software	Mantissa of Sustain rate increment
ExpKp	5	Software	Exponent of Peak rate increment
ManKp	18	Software	Mantissa of Peak rate increment
ExpInvKs	5	Software	Exponent of one over the Sustain rate increment
ManInvKs	18	Software	Mantissa of one over the Sustain rate increment
Shp_thrs	17	Software	Threshold

Figure 424

Field name	#bits	Owned by	Description
Mtr_thrs	27	Software	Number of accumulated cells before flagging as a violator.

Figure 425

Field name	#bits	Owned by	Description
rp_slot	20	Hardware	Read pointer of the FlowID link list.
Wp_slot	20	Hardware	Write pointer of the FlowID link list.
E_slot	1	Hardware	Empty indication for this slot

Figure 426

Field name	#bits	Owned by	Description
Residue	18	Hardware	Residue from input/output phase calculation
Arrival_time	18	Hardware	Indication of the time when a slot arrives to the output FIFO and removed all FlowIDs to the output FIFO. Valid only if start==1.
Addition for dual leaky bucket only:			
last_time	18	Hardware	For the dual bucket. Time of last shaped cell.
Credit	17	Hardware	For dual leaky bucket. Amount of credit accumulated for the FlowID.
peak_sustain	1	Hardware	If set, use peak rate to calculate future slot during output phase. If reset, use sustained rate to calculate future slot during output phase.

Shp_fid1_word_en	Write to field	Memory bits
[0]	Residue	[17:0]
[1]	arrival_time	[35:18]
[2]	last_time	[53:36]
[3]	Credit, d_s	[64:54]

Figure 427

Field name	#bits	Owned by	Description
cell_cnt	27	Hardware	The accumulation of cells between measures.
Mark	1	Hardware	If set, the FlowID violated the assigned rate.
bg_cnt	4	Hardware	Current count of the number of times the background process accessed the FlowID since the last measurement against the threshold.
bg_cnt_value	4	Sotfware	Assigned by the user during setup connection command. The amount of times the FlowID should be accessed by the background process before comparing to the threshold value.

Figure 428

Field name	#bits	Owned by	Description
fid_next	20	Hardware	A pointer to the next FlowID
Addition for single/dual leaky bucket only:			
start	1	Hardware	If set, this FlowID was the head of the link list in one of the slots, used in the output FIFO to replace the time of arrival to all the FlowIDs from this one until the next FlowID that has this bit set.
Addition for dual leaky bucket only:			
bg_stt	2	Hardware	For dual leaky bucket. Set to 11 during setup connection command used to mark the FlowID for time count wrapping around.
time_msb	1	Hardware	Bit [20] (the 21th bit) of the free running time counter. Used for the background process.

Shp_fid2_word_en	Write to field	Memory bits
[0]	fid_next	[19:0]
[1]	start, bg_stt, time_msb	[27:24]

Figure 429

Field name	#bits	Owned by	Description
fid_next	20	Hardware	A pointer to the next FlowID

Figure 430

Address	Name	Type	Description
0	COM	R/W	[31:28] – Opcode [27:0] – Address, depending on the command. No default value.
1	R0	R/W	General-purpose register. No default value
2	R1	R/W	General-purpose register. No default value
3	R2	R/W	General-purpose register. No default value
4	R3	R/W	General-purpose register. No default value
5 – 31	Reserved		
32	CONTROL	R/W	[0] – mode if 1: shaper mode if 0: meter mode Default value: 0 [1] – Shaper output enable if 1: output is enabled for the shaper if 0: output is disabled for the shaper Default value: 0 [2] – Dual/Single leaky bucket If 0, single leaky bucket is supported for all FIDs If 1, duall leaky bucket is supported for all FIDs Default value – 0 [31:3] – Reserved
33	RP_OUT	R/W	[31:21] Reserved [20] – empty indication to the output FIFO. If 1 the output FIFO is empty, no FID for the Output Phase to process. If 0 the output FIFO is not empty Default value: 1 [19:0] Read Pointer for the Output FIFO.
34	WP_OUT	R/W	[31:20] Reserved [19:0] Write Pointer for the Output FIFO.
35	MARK_RP	R/W	[31:21] – Reserved [20] marked link list empty indication for meter block If 1 the output FIFO is empty If 0 the output FIFO is not empty [19:0] Read Pointer for the marked link list
36	MARK_WP	R/W	[31:20] – Reserved [19:0] Write pointer for the marked link list for meter block
37	TEST_REG	R/W	[31:0] – Test mode. TBD
38-63	Reserved		

Figure 431

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0001	R		FID[19:0]				
R0	R[13:0]			Residue[17:0]				
R1	R[13:0]			Arrival_time[17:0]				
R2	R[13:0]			Last_time[17:0]				
R3	R[13:0]			P	Credit[16:0]			

Figure 432

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0010	R		FID[19:0]				
R0	R[13:0]			Residue[17:0]				
R1	R[13:0]			Arrival_time[17:0]				
R2	R[13:0]			Last_time[17:0]				
R3	R[13:0]			P	Credit[16:0]			

Figure 433

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0011	R[7:0]		FID[19:0]				
R0	R[7:0]		SBT	FID_NEXT[19:0]				

Figure 434

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0100	R[7:0]		FID[19:0]				
R0	R[7:0]		SBT	FID_NEXT[19:0]				

Figure 435

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0101	R[9:0]		SLOT_ADDR[17:0]				
R0	R[31:20]		E	RP_SLOT[19:0]				
R1	R[31:20]			WP_SLOT[19:0]				

Figure 436

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0110	R[9:0]		SLOT_ADDR[17:0]				
R0	R[31:20]		E	RP_SLOT[19:0]				
R1	R[31:20]			WP_SLOT[19:0]				

Figure 437

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0111	R				Rateid_addr[9:0]		
R0	R[8:0]		Ks[22:0]					
R1	R[8:0]		Kp[22:0]					
R2	R[8:0]		InvKs[22:0]					
R3	R[14:0]			Shp_threshold[16:0]				

Figure 438

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1000	R				Rateid_addr[9:0]		
R0	R[8:0]		Ks[22:0]					
R1	R[8:0]		Kp[22:0]					
R2	R[8:0]		InvKs[22:0]					
R3	R[14:0]			Shp_threshold[16:0]				

Figure 439

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1001	R	cnt[3:0]	FID[19:0]				

Figure 440

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0001	R		FID[19:0]				
R0	R	MI	Cell_cnt[26:0]					
R1	R						cnt[3:0]	cur[3:0]

Figure 441

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0010	R		FID[19:0]				
R0	R	M	Cell_cnt[26:0]					
R1	R						cnt[3:0]	cur[3:0]

Figure 442

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0011	R		FID[19:0]				
R0	R			Fid_next[19:0]				

Figure 443

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0100	R		FID[19:0]				
R0	R			Fid_next[19:0]				

Figure 444

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0101	R				threshold_addr[9:0]		
R0	R	threshold[26:0]						

Figure 445

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0110	R				threshold_addr[9:0]		
R0	R	threshold[26:0]						

Figure 446

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0111	R		FID[19:0]				
R0	R	MI	Cell_cnt[26:0]					
R1	R						cnt[3:0]	cur[3:0]

Figure 447

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1000	R	cnt[3:0]	FID[19:0]				

Figure 448

Memory	CLK0	CLK1	CLK2	CLK3	CLK4	CLK5	CLK6	CLK7
Meter Input Phase Operations								
RateID (In	RD							
FID1 (Ext	RD			WR				
FID2 (Ext				WR				
Meter Background Phase Operations								
RateID (In								
FID1 (Ext					RD		WR	
FID2 (Ext								
Meter Software Phase Operations (ReadModWrite command)								
RateID (In								
FID1 (Ext						RD		Wr
FID2 (Ext						RD		
CPU Rd/Wr Operations								
RateID (In								*RdWr*
FID1 (Ext								*RdWr*
FID2 (Ext								*RdWr*

Figure 449

#	Block's name	Address [9:6]	Start Address	Last Address	Total length
1	CPU Interface	0000b	0000000b	1111111b	64d
2	Per-Flow Queue	0001b			64d
3	Shaper	0010b			64d
4	Scheduler	0011b			64d
5	DBS	0100b			64d
6 : 16	Reserved	0110b - 1111b			64d : 64d

Figure 450

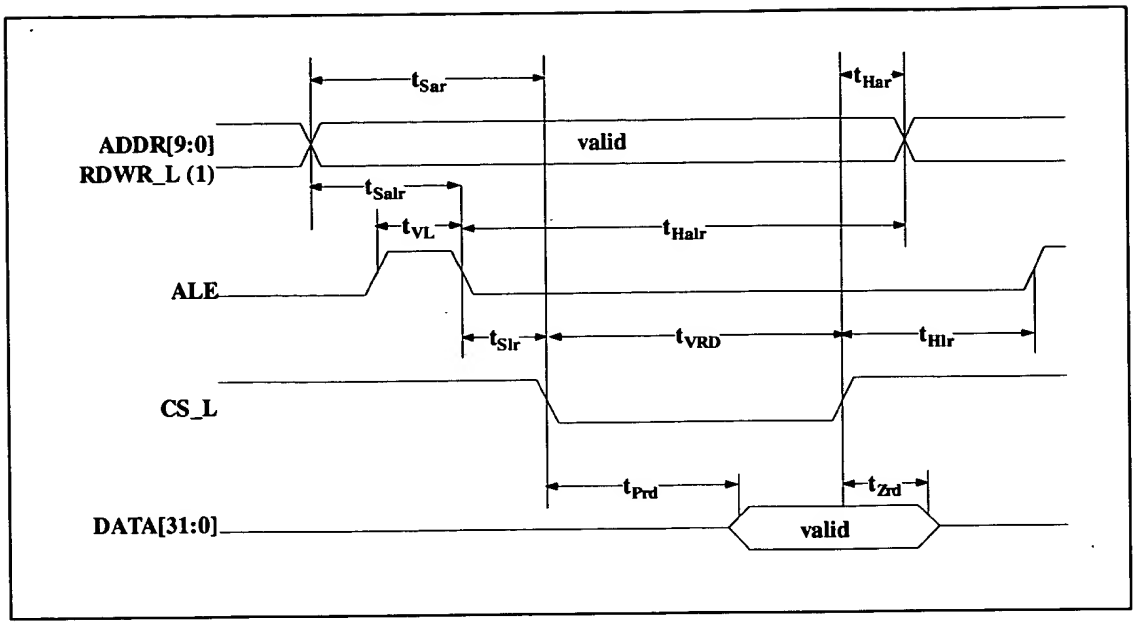


Figure 451

Symbol	Parameter	Min	Max	Units
t_{Sar}	Address/rdwr_l to Valid Read set-up time	0		ns
t_{Har}	Address/rdwr_l to Valid read hold time	0		ns
t_{Salr}	Address to latch set-up time	5		ns
T_{Halr}	Address to latch hold time	5		ns
T_{VL}	Valid latch pulse width	5		ns
T_{Slr}	Latch to Read set-up	0		ns
T_{Hlr}	Latch to Read hold	5		ns
T_{Prd}	Valid Read to valid data propagation delay		50	ns
t_{Zrd}	Valid Read negated to output tri-state		10	ns
t_{VRD}	Valid Read pulse width	60		

Figure 452

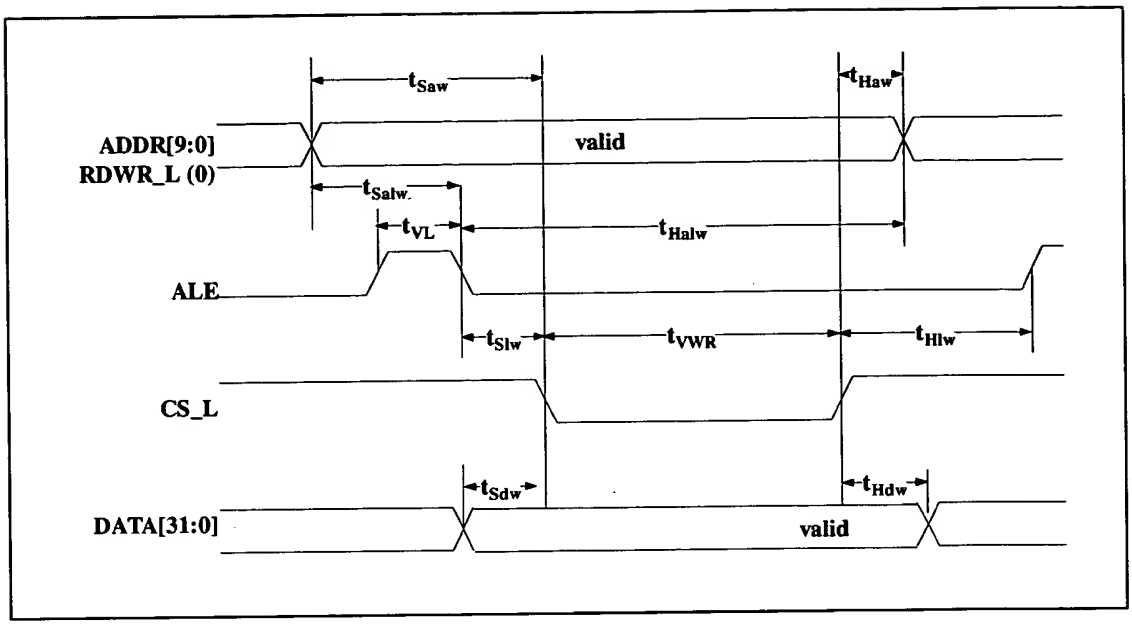


Figure 453

Symbol	Parameter	Min	Max	Units
t_{Saw}	Address/rdwr_l to Valid Write set-up time	0		ns
t_{Haw}	Address/rdwr_l to Valid Write hold time	0		ns
t_{Salw}	Address to latch set-up time	5		ns
T_{Halw}	Address to latch hold time	5		ns
T_{VL}	Valid latch pulse width	5		ns
T_{Slw}	Latch to Write set-up	0		ns
T_{Hlw}	Latch to Write hold	5		ns
T_{Sdw}	Data to valid Write set-up time	0		ns
T_{Hdw}	Data to valid write hold time	5		ns
T_{VWR}	Valid Write pulse width	60		ns

Figure 454

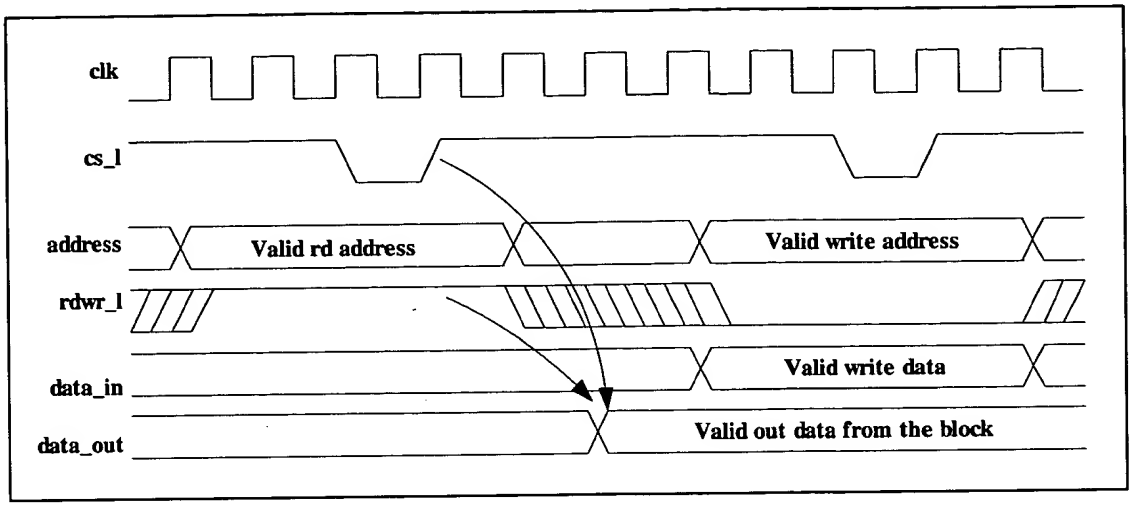


Figure 455

Configuration bit	mode	Description
CPU_CONFIG[0]	Endian select	If reset, big Endian If set, little Endian
CPU_CONFIG[1]	Addr/Data multiplex	If reset, Non-multiplexed If set, Multiplexed

Figure 456

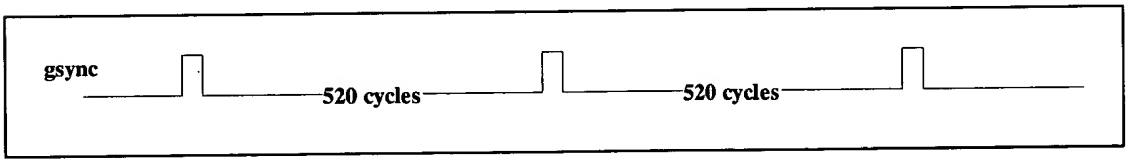


Figure 457

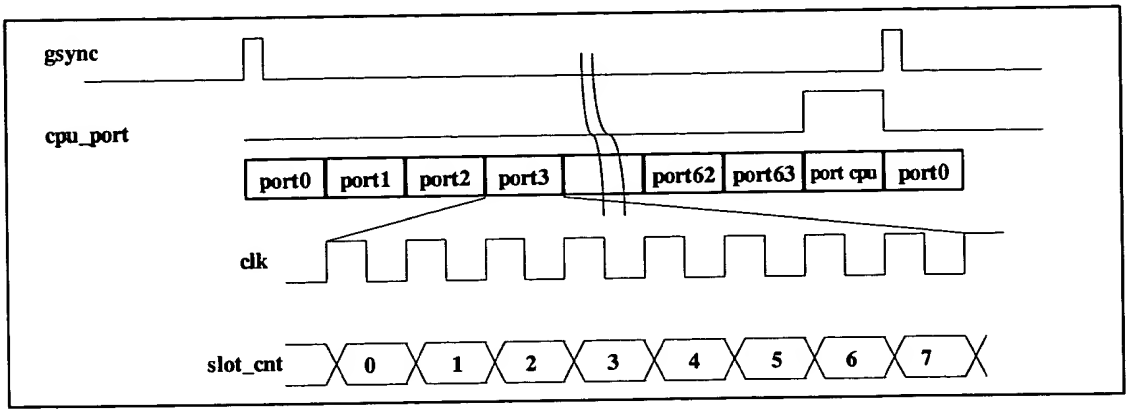


Figure 458

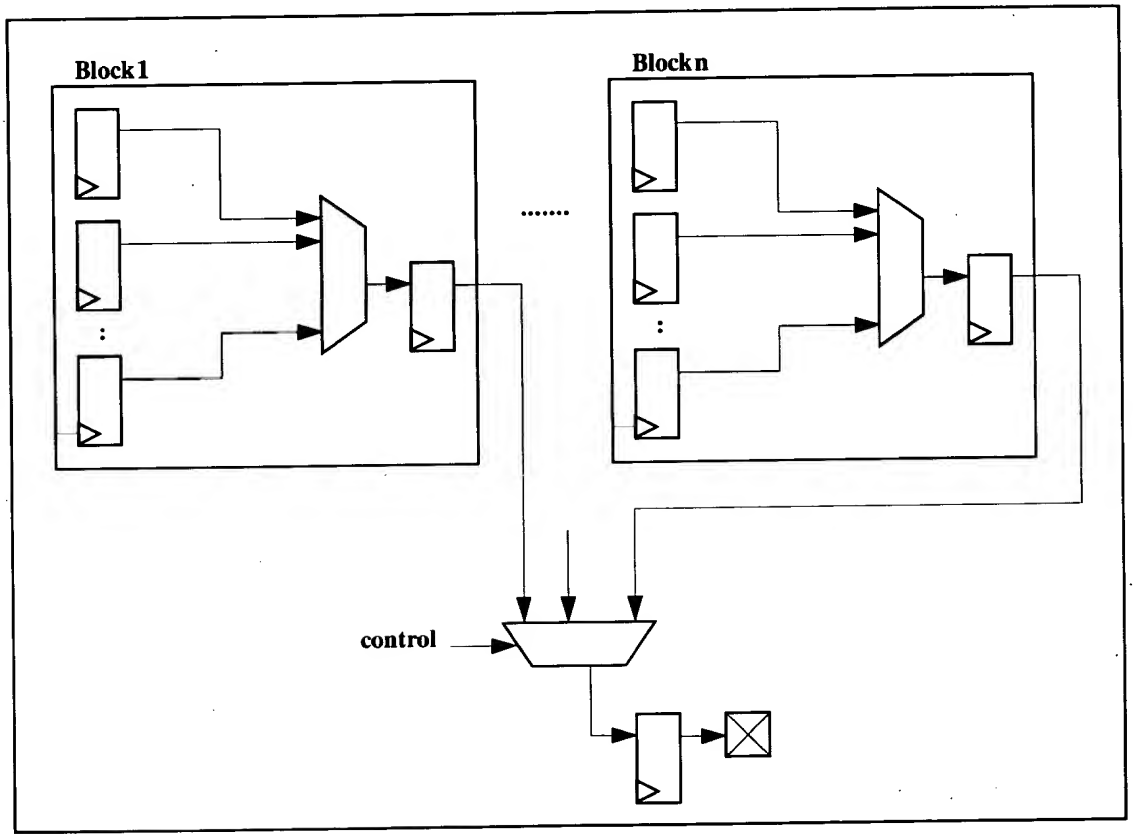


Figure 459

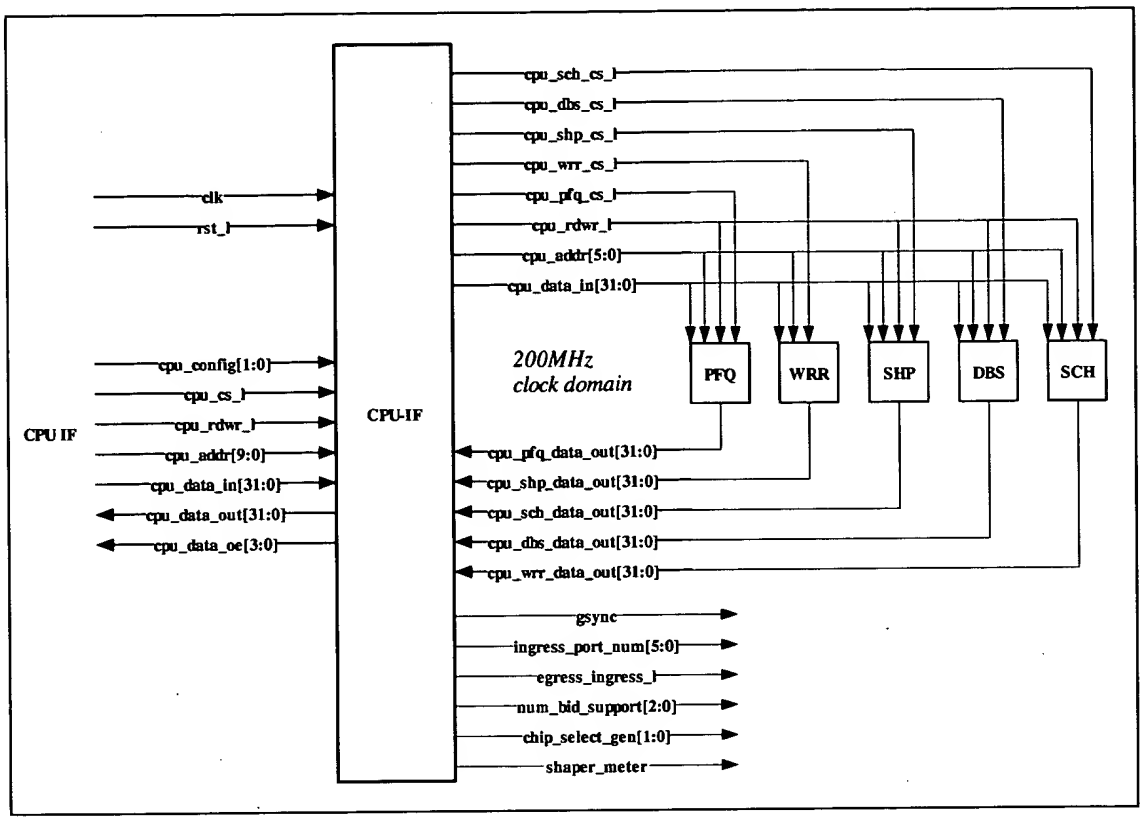

CPU IF
clk
rst_l
cpu_config[1:0]
cpu_cs_l
cpu_rdwr_l
cpu_addr[9:0]
cpu_data_in[31:0]
cpu_data_out[31:0]
cpu_data_oe[3:0]
CPU-IF
cpu_sch_cs_l
cpu_dbs_cs_l
cpu_shp_cs_l
cpu_wrr_cs_l
cpu_pfq_cs_l
cpu_rdwr_l
cpu_addr[5:0]
cpu_data_in[31:0]
200MHz
clock domain
PFQ
WRR
SHP
DBS
SCH
cpu_pfq_data_out[31:0]
cpu_shp_data_out[31:0]
cpu_sch_data_out[31:0]
cpu_dbs_data_out[31:0]
cpu_wrr_data_out[31:0]
gsync
ingress_port_num[5:0]
egress_ingress_l
num_bid_support[2:0]
chip_select_gen[1:0]
shaper_meter

Figure 460

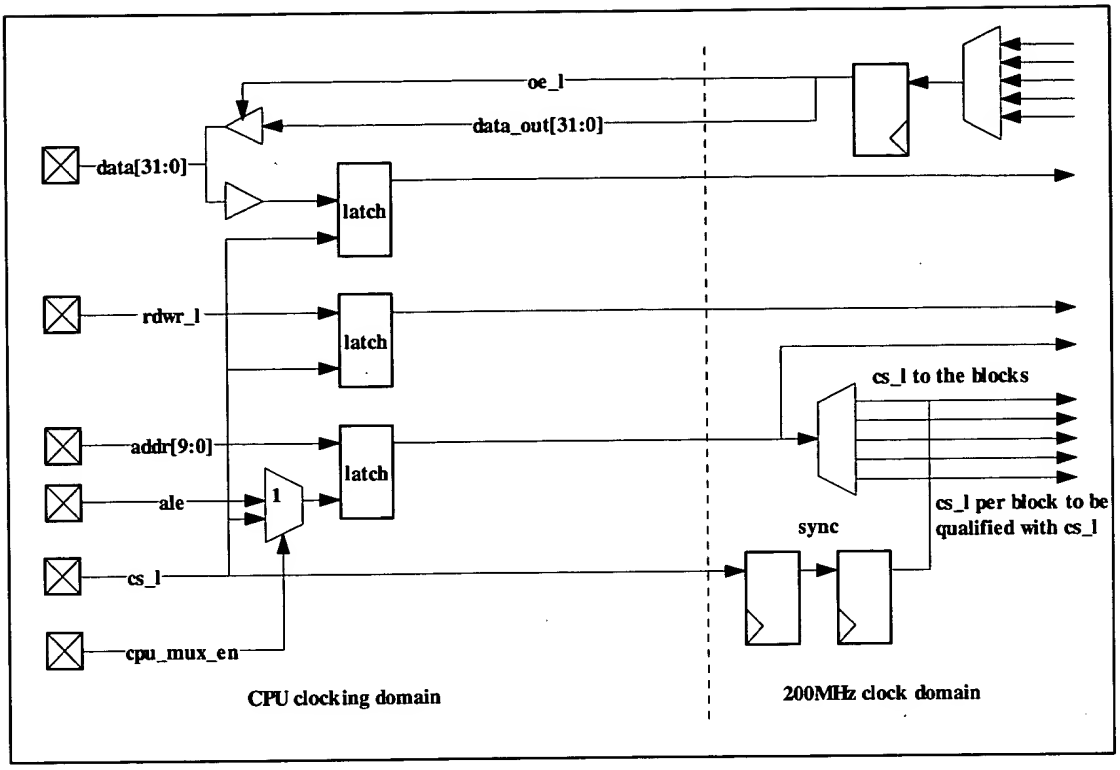

oe_l
data_out[31:0]
data[31:0]
latch
rdwr_l
latch
cs_l to the blocks
addr[9:0]
latch
ale
1
sync
cs_l per block to be qualified with cs_l
cs_l
cpu_mux_en
CPU clocking domain
200MHz clock domain

Figure 461

Address	Name	Type	Description
0	VERSION	RO	Read only register to contains the version of the chip. It's value is 0000_0001h
1	MOD_CTRL	RW	[5:0] – Port number for ingress chip Default value: 0 [6] – Ingress/Egress chip selection 0 – ingress chip 1 – egress chip Default value: 1 [7] – Reserved [8] – Shape/Meter If 0, this block operates as Shaper If 1, this block operates as Meter Default value – 0 [31:9] – Reserved
2	MEM_CTRL	RW	[2:0] – Number of supported BIDs 000 – 1M BIDs 001 – 2M BIDs 010 – 3M BIDs 011 – 4M BIDs 100 – 5M BIDs 101 – 6M BIDs 110 – 7M BIDs 111 – 8M BIDs Default value: 001 [3] – Reserved [5:4] – Chip select generation 00 – based on address[20:19] 01 - based on address[21:20] 10 - based on address[22:21] 11 – Reserved Default value: 00 [31:6] – Reserved

3	RST_CTRL	RW	[3:0] – Reset per block 0000 – no reset 0001 – reset CPU IF block 0010 – reset PFQ block 0011 – reset shaper block 0100 – reset scheduler block 0101 – reset DBS block 0111 – reset all blocks 1000:1111 – Reserved The reset signal to a block is an OR gate between global reset and this indication. This register is cleared by the hardware 8 cycles after writing.
4	TEST_CTRL	RW	[3:0] – Block selection for driving test bus to the pins: 0000 – CPU Interface block 0001 – PFQ block 0010 – Shaper/Meter block 0011 – Scheduler block 0100 – DBS block 0101 – CBWFQ block 0110:1111 - Reserved [31:4] – Reserved
5-63	Reserved		

Figure 462

ADDR[9:0]	address bus, bits[9:6] selects the block, bits [5:0] select the register inside the block.
DATA[31:0]	32 bits of bi-directional data bus
CS_L	Chip select. All CPU transactions are valid when this signal is asserted
RDWR_L	Read/Write indication
ALE	For Multiplexed bus mode. When set, the address is latched

Figure 463

#	Block's name	Address [9:6]	Start Address	Last Address	Total length
1	CPU Interface	0000b	0000000b	1111111b	64d
2	Per-Flow Queue	0001b			64d
3	Shaper	0010b			64d
4	Scheduler	0011b			64d
5	CBWFQ	0100b			64d
6 7 8	Reserved	0101b - 1111b			64d per blcok

Figure 464

Address	Name	Type	Description
0	VERSION	RO	Read only register to contains the version of the chip. It's value is 0000_0001h
1	MOD_CTRL	RW	[5:0] – Port number for ingress chip Default value: 0 [6] – Ingress/Egress chip selection 0 – ingress chip 1 – egress chip Default value: 1 [7] – Reserved [8] – Shape/Meter If 0, this block operates as Shaper If 1, this block operates as Meter Default value – 0 [31:9] – Reserved
2	MEM_CTRL	RW	[2:0] – Number of supported BIDs 000 – 1M BIDs 001 – 2M BIDs 010 – 3M BIDs 011 – 4M BIDs 100 – 5M BIDs 101 – 6M BIDs 110 – 7M BIDs 111 – 8M BIDs Default value: 001 [3] – Reserved [5:4] – Chip select generation 00 – based on address[20:19] 01 - based on address[21:20] 10 - based on address[22:21] 11 – Reserved Default value: 00 [31:6] – Reserved

3	RST_CTRL	RW	[3:0] – Reset per block 0000 – no reset 0001 – reset CPU IF block 0010 – reset PFQ block 0011 – reset shaper block 0100 – reset scheduler block 0101 – reset DBS block 0111 – reset all blocks 1000:1111 – Reserved The reset signal to a block is an OR gate between global reset and this indication. This register is cleared by the hardware 8 cycles after writing.
4	TEST_CTRL	RW	[3:0] – Block selection for driving test bus to the pins: 0000 – CPU Interface block 0001 – PFQ block 0010 – Shaper/Meter block 0011 – Scheduler block 0100 – DBS block 0101 – CBWFQ block 0110:1111 - Reserved [31:4] – Reserved
5-63	Reserved		

Figure 465

Address	Name	Type	Description
0	COM	R/W	[31:28] – Opcode [27:0] – Address, depending on the command. No default value.
1	R0	R/W	General-purpose register. No default value
2	R1	R/W	General-purpose register. No default value
3	R2	R/W	General-purpose register. No default value
4-31	Reserved	NOP	NOT USED
32	TOTAL_FREE_BUFF	R/W	[31:23] – Reserved [22:0] – Total number of Free Buffers Default: 8M – 1 = FFFFE = 1111 1111 1111 1111 1110
33	THRESHOLD_FREE_BUFF	R/W	[31:23] – Reserved [22:0] – Total number of Free Buffers threshold for empty Default: 8M – 11H = FFFEE = 1111 1111 1111 1110 1110

34	THRESHOLD_ FREE_BUFF_BA CK_PRES	R/W	[31:23] – Reserved [22:0] – Total number of Free Buffers threshold for backpressure Default: 8M – 11H = FD8FF = 1111 1101 1000 1111 1111
35	FREE_BUFFS_IN USE	R/W	[31:23] – Reserved [22:0] – Buffers currently in use Loadable counter for testing
36	CONTROL	R/W	[31:4] – Reserved CPU Port blocked [3] if set the CPU port is Blocked. Default value is "1". Enqueue Multicast as Unicast traffic [2] If set the multicast traffic is treated as Unicast. Default value "0". Enable buffer management [1] If set the buffer management is enabled. Default value "0". RED/CLASS [0] If set the buffer management is RED. Default value "0".
37	RED TIME Q-TIME	R/W	[31:25] – Reserved, Counter [24:0] - Count
38	Transmission time "S"	R/W	[31:16] Reserved, [15:0] Typical transmission time for a small packet.
39	FID Memory descriptor	R/W	[31:20] Reserved, [19:0] MASK contiguously used. Number of bits used to access the FID memory. The number of buffers can be less than the available storage. Default value "FFFFF"
40	Timeout Rate	R/W	[31:5] Reserved, [4:0] Timeout rate. It is the number of clocks to skip before the next timeout. Default value is "00000".
41	FREE Buffer Tail	R/W	Free Buffer tail [22:0]
42	FREE Buffer Head	R/W	Free Buffer Head [22:0]
43	OUTPUT PORT BLOCKED [31:0]	R/W	[31:0] Output ports 31 downto 0 statuses. If set, then the port is blocked and discard command with EOP cell is asserted.
44	OUTPUT PORT BLOCKED [63:32}	R/W	[31:0] Output ports 63 downto 32 statuses. If set, then the port is blocked and discard command with EOP cell is asserted.
45	CLASS 0 Threshold	R/W	[23:0] CLASS 0 buffer management Threshold. Default value "000FFF".
46	CLASS 1 Threshold	R/W	[23:0] CLASS 1 buffer management Threshold. Default value "000FFF".
47	CLASS 2 Threshold	R/W	[23:0] CLASS 2 buffer management Thresholds. Default value "000FFF".

48	CLASS 3 Threshold	R/W	[23:0] CLASS 3 buffer management Thresholds. Default value "000FFF".
49	CLASS 4 Threshold	R/W	[23:0] CLASS 4 buffer management Thresholds. Default value "000FFF".
50	CLASS 5 Threshold	R/W	[23:0] CLASS 5 buffer management Thresholds. Default value "000FFF".
51	CLASS 6 Threshold	R/W	[23:0] CLASS 6 buffer management Thresholds. Default value "000FFF".
52	CLASS 7 Threshold	R/W	[23:0] CLASS 7 buffer management Thresholds. Default value "000FFF".
53	CLASS 0 Counter	R/W	[23:0] CLASS 0 buffer management Counter.
54	CLASS 1 Counter	R/W	[23:0] CLASS 1 buffer management Counter.
55	CLASS 2 Counter	R/W	[23:0] CLASS 2 buffer management Counter.
56	CLASS 3 Counter	R/W	[23:0] CLASS 3 buffer management Counter.
57	CLASS 4 Counter	R/W	[23:0] CLASS 4 buffer management Counter.
58	CLASS 5 Counter	R/W	[23:0] CLASS 5 buffer management Counter.
59	CLASS 6 Counter	R/W	[23:0] CLASS 6 buffer management Counter.
60	CLASS 7 Counter	R/W	[23:0] CLASS 7 buffer management Counter.
61	TEST_REG	R/W	Test mode
62 - 63	Reserved	NOP	NOT USED

Figure 466

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0001	Don't Care		FID_ADDR [20:0]				
R0	R0 [31:0] Data; Written by the PFQ and Read by the CPU							
R1	R1 [31:0] Data; Written by the PFQ and Read by the CPU							
R2	R2 [7:0] Data; Written by the PFQ and Read by the CPU							

Figure 467

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0010	Don't Care		FID_ADDR [20:0]				
R0	R0 [31:0] Data; Written by the CPU							
R1	R1 [31:0] Data; Written by the CPU							
R2	R2 7:0] Data; Written by the CPU							

Figure 468

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0011	Don't Care		FID_ADDR [19:0]				
R0	R0 [31:0] Data; Written by PFQ and Read by CPU							
R1	R0 [3:0] Data; Written by PFQ and Read by CPU							

Figure 469

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0100	Don't Care		FID_ADDR [19:0]				
R0	R0 [31:0] Data; Written by the CPU							
R1	R1 [3:0] Data; Written by the CPU							

Figure 470

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0101	BID_ADDR [22:0]						
R0	R0 [31:0] Data; Written by the PFQ and Read by the CPU							
R1	R1 [3:0]; Written by the PFQ and Read by the CPU							

Figure 471

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0110	BID_ADDR [22:0]						
R0	R0 [31:0] Data; Written by the CPU							
R1	R1 [3:0]; Written by the CPU							

Figure 472

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0111	Don't Care		FID_ADDR [20:0]				
R0	R0 [31:0] Data; Written by the PFQ and Read by the CPU							
R1	R1 [31:0] Data; Written by the PFQ and Read by the CPU							
R2	R2 [7:0] Data; Written by the PFQ and Read by the CPU							

Figure 473

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1000	Don't Care		FID_ADDR [20:0]				
R0	R0 [31:0] Data; Written by the CPU							
R1	R1 [31:0] Data; Written by the CPU							
R2	R2 [7:0] Data; Written by the CPU							

Figure 474

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1001	Don't Care		FID_ADDR [19:0]				
R0	RED ASSOCIATION [9:0] 9:0 PORT [5:0] 15:10							

Figure 475

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1010	Don't Care		FID_ADDR [19:0]				

Figure 476

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1011	Don't Care						
R0	Don't Care							
R1	Don't Care							

Figure 477

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1100	Don't Care						
R0	Don't Care							
R1	Don't Care							

Figure 478

Address	Name	Type	Description
0	COM	R/W	[31:28] – Opcode [27:0] – Address, depending on the command. No default value.
1	R0	R/W	General-purpose register. No default value
2	R1	R/W	General-purpose register. No default value
3	R2	R/W	General-purpose register. No default value
4	R3	R/W	General-purpose register. No default value
5 – 31	Reserved		
32	Control	R/W	[0] – Modify CLP enable If reset, no modification is allowed to the CLP bit If set, CLP bit can be modified according to shapers outputs. Default value: 0 [31:1] – Reserved
33	TEST_REG	R/W	[31:0] – Test mode. TBD
34-63	Reserved		

Figure 479

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0001	R		FID[19:0]				
R0	R[7:0]		S Pri	R	Qos[9:0]		A	Port[5:0]
R1	R[7:0]		rp[23:0]					
R2	R[9:8]		wp[23:0]					
R3	R[9:0]			Cell_cnt[9:0]		C	Ncell[9:0]	

Figure 480

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0010	R		FID[19:0]				
R0	R[7:0]		S Pri	R	Qos[9:0]		A	Port[5:0]
R1	R[7:0]		rp[23:0]					
R2	R[7:0]		wp[23:0]					
R3	R[9:0]			Cell_cnt[9:0]		C	Ncell[9:0]	

Figure 481

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0011	R		FID[19:0]				
R0	R		S Pri	R	Qos[9:0]		A	Port[5:0]

Figure 482

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0100	R	Pkt[23:0]					
R0	Fid[19:0]					L	Ncell[9:0]	
R1	R		Next[23:0]					

Figure 483

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0101	R	Pkt[23:0]					
R0	Fid[19:0]					L	Ncell[9:0]	
R1	R		Next[23:0]					

Figure 484

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0110	R			Data[7:0]		Port_addr[7:0]	

Figure 485

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0111	R			Data[7:0]		Port_addr[7:0]	

Figure 486

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1000	R		P[3:0]	Empty[7:0]		R	port_add[5:0]

Figure 487

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1001	R		P[3:0]	Empty[7:0]		R	port_add[5:0]

Figure 488

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1010	R					class_addr[8:0]	
R0	R		E	rp[23:0]				
R1	R			wp[23:0]				

Figure 489

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1011	R					class_addr[8:0]	
R0	R		E	rp[23:0]				
R1	R			wp[23:0]				

Figure 490

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1100	R						

Figure 491

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1101	R						

Figure 492

Address	Name	Type	Description
0	COM	R/W	[31:28] – Opcode [27:0] – Address, depending on the command. No default value.
1	R0	R/W	General-purpose register. No default value
2	R1	R/W	General-purpose register. No default value
3	R2	R/W	General-purpose register. No default value
4	R3	R/W	General-purpose register. No default value
5 – 31	Reserved		
32	CONTROL	R/W	[0] – mode if 1: shaper mode if 0: meter mode Default value: 0 [1] – Shaper output enable if 1: output is enabled for the shaper if 0: output is disabled for the shaper Default value: 0 [2] – Dual/Single leaky bucket If 0, single leaky bucket is supported for all FIDs If 1, duall leaky bucket is supported for all FIDs Default value – 0 [31:3] – Reserved

33	RP_OUT	R/W	[31:21] Reserved [20] – empty indication to the output FIFO. If 1 the output FIFO is empty, no FID for the Output Phase to process. If 0 the output FIFO is not empty Default value: 1 [19:0] Read Pointer for the Output FIFO.
34	WP_OUT	R/W	[31:20] Reserved [19:0] Write Pointer for the Output FIFO.
35	MARK_RP	R/W	[31:21] – Reserved [20] marked link list empty indication for meter block If 1 the output FIFO is empty If 0 the output FIFO is not empty [19:0] Read Pointer for the marked link list
36	MARK_WP	R/W	[31:20] – Reserved [19:0] Write pointer for the marked link list for meter block
37	TEST_REG	R/W	[31:0] – Test mode. TBD
38-63	Reserved		

Figure 493

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0001	R		FID[19:0]				
R0	R[13:0]			Residue[17:0]				
R1	R[13:0]			Arrival_time[17:0]				
R2	R[13:0]			Last_time[17:0]				
R3	R[13:0]			P	Credit[16:0]			

Figure 494

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0010	R		FID[19:0]				
R0	R[13:0]			Residue[17:0]				
R1	R[13:0]			Arrival_time[17:0]				
R2	R[13:0]			Last_time[17:0]				
R3	R[13:0]			P	Credit[16:0]			

Figure 495

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0011	R[7:0]		FID[19:0]				
R0	R[7:0]		SBT	FID_NEXT[19:0]				

Figure 496

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0100	R[7:0]		FID[19:0]				
R0	R[7:0]		SBT	FID_NEXT[19:0]				

Figure 497

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0101	R[9:0]		SLOT_ADDR[17:0]				
R0	R[31:20]		E	RP_SLOT[19:0]				
R1	R[31:20]			WP_SLOT[19:0]				

Figure 498

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0110	R[9:0]		SLOT_ADDR[17:0]				
R0	R[31:20]		E	RP_SLOT[19:0]				
R1	R[31:20]			WP_SLOT[19:0]				

Figure 499

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0111	R				Rateid_addr[9:0]		
R0	R[8:0]		Ks[22:0]					
R1	R[8:0]		Kp[22:0]					
R2	R[8:0]		InvKs[22:0]					
R3	R[14:0]			Shp_threshold[16:0]				

Figure 500

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1000	R				Rateid_addr[9:0]		
R0	R[8:0]		Ks[22:0]					
R1	R[8:0]		Kp[22:0]					
R2	R[8:0]		InvKs[22:0]					
R3	R[14:0]			Shp_threshold[16:0]				

Figure 501

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1001	R	cnt[3:0]	FID[19:0]				

Figure 502

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0001	R		FID[19:0]				
R0	R	MI	Cell_cnt[26:0]					
R1	R						cnt[3:0]	cur[3:0]

Figure 503

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0010	R		FID[19:0]				
R0	R	M	Cell_cnt[26:0]					
R1	R						cnt[3:0]	cur[3:0]

Figure 504

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0011	R		FID[19:0]				
R0	R			Fid_next[19:0]				

Figure 505

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0100	R		FID[19:0]				
R0	R			Fid_next[19:0]				

Figure 506

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0101	R				threshold_addr[9:0]		
R0	R	threshold[26:0]						

Figure 507

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0110	R				threshold_addr[9:0]		
R0	R	threshold[26:0]						

Figure 508

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0111	R		FID[19:0]				
R0	R	MI	Cell_cnt[26:0]					
R1	R						cnt[3:0]	cur[3:0]

Figure 509

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1000	R	cnt[3:0]	FID[19:0]				

Figure 510

Address	Name	Type	Description
0	COM	R/W	[31:28] – Opcode [27:0] – Address, depending on the command. Default value for bits [31:28] – 0 No default value for bits [27:0]
1	R0	R/W	General-purpose register. No default value
2	R1	R/W	General-purpose register. No default value
3	R2	R/W	General-purpose register. No default value
4 – 31	Reserved		
32	CONTROL	R/W	[0] – OUT_EN Global output enable for all ports, if reset, no output stage will be performed for all ports Default value 0 (output disabled). [31:1] – Reserved
33	INGRESS_PTR	R/W	[7:0] – Head pointer of the virtual port list. Used only in case of an ingress chip. No default value. [15:8] – Tail pointer of the virtual port list. Used only in case of an ingress chip. No default value. [16] – Empty indication of the virtual port list. Used only in case of an ingress chip. If set, the virtual port list is empty. Default value 1 (empty list) [31:17] – Reserved.
34	CPU_R_PTR	R/W	[19:0] – Head pointer to the FlowID list of the CPU port. No default value.
35	CPU_W_PTR	R/W	[19:0] – Tail pointer to the FlowID list of the CPU port. No default value.

36	CPU_CTRL	R/W	[0] – Empty indication of the CPU FlowID list. If set, the FlowID linked list is empty. Default value 1 (empty list) [1] – CPU output port enable. If set, the scheduler can schedule FlowIDs for the CPU port. Default value 0 (Disable the CPU port) [31:2] – Reserved
37	WEIGHT_QUOTA	R/W	[15:0] – weight_quota This value is a multiplicand to calculate the weight per QOS. Default value 1 [31:16] – Reserved
38	TEST_REG	R/W	[31:0] – Test mode. TBD
39 - 63	Reserved		

Figure 511

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0001	R		Fid[19:0]				
R0	R			Next[19:0]				

Figure 512

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0010	R		Fid[19:0]				
R0	R			Next[19:0]				

Figure 513

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0011	R[17:0]				Addr[9:0]		
R0	R[12:0]			qnum	Weight		R[1:0]	PortID

Figure 514

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0100	R[17:0]				Addr[9:0]		
R0	R[12:0]			qnum	Weight		R[1:0]	PortID

Figure 515

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0101	R[17:0]					Addr[9:0]	
R0	R[10:0]			E	Read PTR[19:0]			
R1	R[11:0]				Write PTR[19:0]			
R2	R[6:0]	W_M[24:0]						

Figure 516

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0110	R[17:0]					Addr[9:0]	
R0	R[10:0]			E	Read PTR[19:0]			
R1	R[11:0]				Write PTR[19:0]			
R2	R[6:0]	Q_WEIGHT_FM[24:0]						

Figure 517

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0111	R[21:0]						Addr[5:0]
R0	Priority[7:0]		R[4:0]	Facto r[2:0]	R[5:0]		Q_B_Addr[9:0]	

Figure 518

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1000	R[21:0]						Addr[5:0]
R0	Priority[7:0]		R[4:0]	Facto r[2:0]	R[5:0]		Q_B_Addr[9:0]	

Figure 519

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1001	R[21:0]						Addr[5:0]
R0	R[8:0]			PQ[6:0]		Qa_Empty[7:0]		Qw_Empty[7:0]
R1	R[7:0]	Q_WEIGHT_FM[23:0]						

Figure 520

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1010	R[21:0]						Addr[5:0]
R0	R[8:0]			PQ[6:0]		Qa_Empty[7:0]		Qw_Empty[7:0]
R1	R[7:0]	Q_WEIGHT_FM[23:0]						

Figure 521

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1011	R[21:0]					Addr[5:0]	
R0	R[25:0]						Next[5:0]	

Figure 522

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1100	R[21:0]					Addr[5:0]	
R0	R[25:0]						Next[5:0]	

Figure 523

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	0100	R[17:0]				Addr[9:0]		
R0	R[12:0]			qnum	Weight[7:0]		PortID[7:0]	

Figure 524

Register	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
COM	1101	R[21:0]					Addr[5:0]	
R0	R[5:0]	Q_B_Addr[9:0]			Priority[7:0]		Factor[7:0]	

Figure 525

#	Block	Parameter	Size [Bits]	Description
1	All		20	Flow ID to setup
	PFQ			
	DBS	Port	6	The value of the output port the FID belongs to (not including the CPU port)
	DBS	CPU_PORT	1	If set, the FID belongs to the CPU port (ignore the PORT field). If reset, the FID belongs to the port defined by the PORT field
	DBS	SHAPE	1	If set, the FID has to be shaped If reset, the FID is not to be shaped (it will be scheduled in the scheduler block)
	DBS	QOS/RATE	10	If SHAPE is set, the field represent the location of the rate to be shaped in the RateID memory inside the Shaper. If SHAPE is reset, the field represent the location of the QOS in the QOS memory inside the scheduler.
	DBS	SHAPE_ CLASS	3	Used only if SHAPE bit is set. This field defines the priority of the FID during the output phase from he shaper.
	SHP	BG_CNT	4	Used only if SHAPE bit is set. This field defines the amount of times the background process of the Meter should count.

Figure 526

#	Block	Name		8M cells, 1M FIDs		4M cells, 1M FIDs		2M cells, 1M FIDs	
				#devices 512Kx36	#devices 1Mx36	#devices 512Kx36	#devices 1Mx36	#devices 512Kx36	#devices 1Mx36
1	PFQ	EnQueue	2Mx72	8	4	8	4	8	4
		DeQueue	1Mx36	2	1	2	1	2	1
		BID	8Mx36	16	8	8	4	4	2
		Statistics	2Mx72	8	4	8	4	8	4
2	SHAPER	FID1	1Mx72	4	2	4	2	4	2
		FID2	1Mx24	2	1	2	1	2	1
		SLOT	256Kx41	2	2	2	2	2	2
3	DATABASE	FID	1Mx91	6	3	6	3	6	3
		PKT	8Mx55	32	16	16	8	8	4
4	SCHEDULER	FID	2Mx25	4	2	4	2	4	2
	total mem units			**84**	**43**	**60**	**31**	**48**	**25**

Figure 527